The Application of Artificial Intelligence Techniques to the Deep-Sea Container-Ship Cargo Stowage Problem

by

Ian David Wilson BSc (Hons)

A thesis submitted in partial fulfilment of the requirements of the University of Glamorgan/ Prifysgol Morgannwg for the degree of Doctor of Philosophy.

Jointly sponsored by The Engineering and Physical Sciences Research Council and The Maritime Computer and Technical Services, Sengenydd Road, Cathays, Cardiff.

School of Accounting and Mathematics

Division of Mathematics and Computing

The University of Glamorgan

May 1997

Table of Contents

TABLE OF FIGURES

Figures

TABLE OF TABLES

Acknowledgements

I wish to express sincere thanks to my supervisors at the University of Glamorgan, Dr. P. Roach, Dr. R. W. Williams and Dr. J. A. Ware, to Mr. G. Ross my supervisor at The Maritime Computer and Technical Services (MCTS), Cardiff, and to Dr. I. Inglis for the encouragement, guidance and constructive criticism they have provided during this research project.

Thanks are also extended to Mr. C. J. Willis, stowage co-ordinator for P&O Containers Ltd. London, for his assistance and to Dr. D. K. Roach for detailed stowage planning information.

Certificate of Research

This is to certify that, except where specific reference is made, the work presented in this thesis is the result of the investigation undertaken by the candidate.

Candidate ....I. D. Wilson..........

Director of Studies ....Paul Roach..........

Declarations

This is to certify that neither this thesis or any part of it has been presented or is being currently submitted in candidature for any other degree other than that of Doctor of Philosophy of the University of Glamorgan/Prifysgol Morgannwg.

Candidate I.D. Wilson

The Application of Artificial Intelligence Techniques to the Deep-Sea Container-Ship Cargo Stowage Problem

by

Ian David Wilson BSc. (Hons.)

A thesis submitted in partial fulfilment of the requirements of the University of Glamorgan/ Prifysgol Morgannwg for the degree of Doctor of Philosophy.

Jointly sponsored by The Engineering and Physical Sciences Research Council and Maritime Computer and Technical Services, Cardiff.

School of Accounting and Mathematics

Division of Mathematics and Computing

The University of Glamorgan

May 1997

ABSTRACT

Container-ships are vessels possessing an internal structure that facilitates the handling of containerised cargo. At each port along the journey of a container-ship, containers destined for that port are unloaded, and some containers destined for subsequent ports are loaded. Determining a viable configuration of containers that facilitates this unloading and loading, in a cost-effective way, constitutes the deep-sea container-ship stowage problem. The work of determining a stowage configuration for a container-ship, on leaving a port, is performed by human stowage planners. The success of a configuration requires consideration of many factors. These factors include whether the configuration contravenes ship stability, minimises the physical costs of handling the containers, and takes into account expected container loads at subsequent ports. Further complications arise from the existence of hazardous cargo that must be segregated from other cargo and the ship's crew, and from the need to handle containers of non-standard dimensions. Stowage planners must work under strict time constraints, and are limited in the number of stowage configurations that they can consider. This real-world problem seems to be one that would benefit from automation through the application of artificial intelligence.

Although many decision support systems exist that automate the time-consuming calculations for ship stability, little work has been published in the area of full automation of stowage planning. Authors proposing full automation have correctly identified the salient features of the problem, but have allowed the array-like nature of spaces within containerised vessels to entirely dictate their approach to addressing the placements of specific containers to specific locations. To enable the implementation of these approaches, excessively large search spaces are pruned through the removal of important features of the problem, rendering the solutions not commercially viable. By concentrating solely on the specific placements of containers, these authors have not recognised how human planners solve the problem. The author of this thesis approaches the container-ship stowage problem from a knowledge engineer's perspective. In the proposed approach, 'intelligence' is provided through the application of the findings of a knowledge elicitation exercise and a systems analysis of the work of human planners. The assumed heuristics inherent in their use of documents are highlighted. This thesis reports on the results of the analysis of the processes employed by a stowage planner. Explanations are provided of how these results allow the problem to be decomposed into sub-problems. An implementation of the approach described would determine good, although not necessarily optimum, solutions to the entire problem in a commercially viable duration of time. Further, this approach allows many more stowage configurations to be considered than would be possible for a human planner. The work contained within this thesis demonstrates the feasibility and benefits of such an implementation. The last chapter contains, in addition to a full and detailed list of conclusions made during the research, a summary of some of those areas that still require further work.

1 INTRODUCTION

1.1 Background to the container-ship stowage problem

Since the 1970s, *containerisation* (the packing of cargo into large, dedicated boxes enabling multiple units of cargo to be handled simultaneously) of cargo transportation has been the norm in the world-wide maritime services. Shipping companies compete around the world to provide profitable, cost effective *container* (essentially a box that comes in a variety of dimensions and types that facilitates the transportation of cargo) transportation services. In order to increase the benefits of economy of scale, the size of *container-ships* has increased. The term used to indicate how many containers of a standard height and width, twenty feet in length, a container-ship can carry is twenty foot equivalent units (TEU). The increase in capacity has been typically from relatively small 350 TEU to ships with capacities of more than 2500 TEU's. This trend appears to be continuing with 4000-6000 TEU container-ships entering service. The introduction of these new dedicated, *cellular* (the term used to indicate that the ship has an internal structure that facilitates the handling of standard length containers) container-ships is due to the success of container standardisation. This standardisation of containers has permitted the introduction of *inter-modal* transportation systems. That is, containerised cargo can be transported by rail, truck or sea due to its standard frame and dimensions having enabled the introduction of carriers dedicated to this purpose. (Further details on container standardisation can be found in Appendix C.)

Container transportation by sea involves the interaction of two, principal, commercial bodies, the *container-terminal operator* (who is responsible for providing the logistics required to load, unload and process containers onto land links) and the *shipping operator* (who is responsible for transporting the containers by sea). The operator of a *container-terminal* (a port configured for the handling of containers) is interested in maximising the throughput of containers. The shipping operator is interested in minimising the time a container-ship spends in port. These two economic considerations often conflict.

Therefore, it is important that the *loading* (the process whereby containers are loaded onto the container ship) and *discharging* (the process whereby containers are unloaded from the container ship) of container ships be carried out with a minimum of disruption.

The large container-ships of today can require thousands of container *movements* per port of call (the loading, unloading or re-positioning of each container) to complete the discharge and load process. Given the large number of container movements associated with modern container ships, reaching optimum efficiency is very difficult. For the shipping operation to be cost-effective it is essential that *vessel utilisation* (the use of cargo space usually expressed as a percentage of TEU capacity) be maximised. This requirement makes the perfect *stowage* (the placement of cargo within the container-ship) of containers difficult, if not impossible since, as the ship fills, fewer stowage alternatives are available to the person planning the layout.

Container-terminal and container-ship efficiency are largely determined by the arrangement of containers both within the container-terminal and on the container-ship. Determining the arrangement of these containers is an error prone process relying largely on the intuitive skills of individual specialists called *planners* (a generic term used to describe the individual(s) responsible for planning the stowage of cargo).

The planner must determine the optimum placement of containers so that all *constraints* (restrictions placed upon where and how containers can be stowed) are satisfied and *material handling costs* (the costs associated with loading, unloading and transporting cargo) are minimised. One of the most important problems associated with this optimisation process is the *re-handle*. A re-handle is a container movement that requires a container to be moved only in order to permit access to another, and is considered to be the result of poor planning (explained in detail in Section 3.2.2). Minimising the number of re-handles, both those associated with a container-ship and those that occur within the container-terminal, reduces operating costs and helps maximise container-terminal efficiency. The planner's task is split into two main parts, the generation of long term, generalised and short term, specialised stowage strategies. These are, forthwith, termed by the author as the *strategic* and *tactical* phases of the container stowage planning activity. Acquiring the knowledge required to be a truly effective planner can take many years. This knowledge is only partly explicit, the true expertise being almost wholly intuitive.

The problems involved with planning are magnified by the multi-port nature of container-ship loading. A plan for a stowage configuration at one port must take into account the consequences (in terms of re-handles, for example at subsequent ports). Determining a stowage configuration for a container-ship, on leaving a port, which is cost-effective constitutes the deep-sea container-ship stowage problem.

1.2 The use of computers to assist planning

In the main, the use of computers to aid the container stowage planning process has focused upon assisting planners with the *tactical* (specific placement of individual containers) phase of the planning operation where the focus has been upon assisting with the conceptualisation of a myriad of different stowage patterns through interactive computer displays where the program calculates the stress and stability of the vessel. These computer applications have been implemented since this is where immediate savings in time and effort could, and have, been made. [12] However, the extent of this improvement is still constrained by the effectiveness of the planner. The keen insight of the planner, supported by experience and knowledge, remains the most important factor in the success of the planning task.

Container terminal planners are beginning to see the benefit of computerised assistance. The computer applications becoming available to the planner of container-terminals assist in improving efficiency resulting in a reduction of operating costs. The same is not true for the planners of container-ships. Whereas considerable attention has been given to automating container-terminal processes, very little attention has been given to automating the optimisation of the stowage strategy of the container-ship.

Until recently, computerised assistance for shipping-line planners in addressing the stowage problem has been limited to the presentation and easy manipulation of stowage plans and the corresponding calculation of vessel stress and stability. However, even the degree of automation offered by these stowage *tools* has not been universally adopted. Planning is still largely carried out by hand, thus making the strategic and tactical planning of cargo stowage a very costly, difficult and often error prone-operation to perform.

1.3 Purpose and scope of study

Stowage planning tools make little use of expertise since they are designed for use by informed, experienced personnel. Capturing this expertise and augmenting it with the computational power of modern computers will ultimately make the whole planning task more effective. This may be achieved by use of an *expert system* (a computer application that is imbued with a degree of expertise in a domain that enables it to emulate a human domain expert), which is one of the practical applications of artificial intelligence technology.

An expert system should provide valuable information that supports judgements and acts as a repository for accumulated experience and knowledge about particular operations. The potential use of expert systems for solving maritime related problems, such as container allocation planning (the world wide distribution of containers effected to streamline cargo transportation), has generated a large amount of interest in the maritime community. [1]

The goal of cargo transport planning research is to develop an automated method of generating stowage patterns and container loading/unloading sequences of placements that will meet all the specific conditions while minimising operating costs. Container transportation research should reflect the views of the ship owner, who is concerned with maximising utilisation of cargo space while minimising operating costs, and that of the container-terminal manager, who is concerned with maximising the quantity of containers passing through the terminal whilst minimising operating costs.

Past attempts by other researchers to automate the tactical and strategic planning procedures have met with varying degrees of success. Computer applications exist that support the container-terminal operations. [2,8,9,10,11] Similarly, software that supports the container-ship stowage planner is beginning to filter into the market. [76] However, little has been accomplished when dealing with the strategic planning phase of the operation. The aim of this project was to bridge the gap between the automation of the terminal operation and that of the shipping company's processes. Therefore, this work investigates the application of artificial intelligence techniques to both the strategic and tactical planning processes.

1.4 Thesis Outline

Chapter 2 provides an overview of the deep-sea container transportation process including detailed discussion about important considerations such as:

- a summary of different containers type;
- an introduction to different container-ships types;
- a summary of the different types of cargo transported;
- an introduction into how container stowage locations are labelled;
- a brief summary of salient container-terminal features.

Chapter 3 provides a detailed discussion about the container stowage planning process and what considerations must be dealt with by the human planner. An introduction to Artificial Intelligence (A.I.) theory and implementation is given in Chapter 4. Chapter 5 provides detailed discussions of relevant existing work in the area of computer aided and automated ship loading. Special consideration is given to the strengths and weaknesses attributed to this work. Chapter 6 provides an outline of the important features required in an automated deep-sea container-ship planning system, with the design process used by the author for a proposed system being outlined in Chapter 7. A detailed discussion on the proposed automated planning system is then given in Chapter 8. Finally, Chapter 9 provides a summary of the conclusions reached by the author on the problem domain, the system analysis exercise and the proposed planning system. Recommendations for future work are also made in that chapter.

2 MARINE TRANSPORTATION OF CONTAINERISED CARGO

This chapter outlines the issues that relate to the transportation of containerised cargo by sea, and which will be important for an understanding of the domain of the deep-sea container-ship stowage problem. To this end the following areas are described:

- the different types of container;
- the variety of cargo carried;
- the types and layout of container-ships;
- the problems caused by cargo placement within a ship;
- how container-terminals are organised;
- how information is transmitted between container-terminals and shipping lines.

Reference is made throughout to sources of more detailed information.

2.1 The Container

This section provides information about:

- the different types and sizes of container;
- the variety of cargo that different containers can carry ;
- the constraints on containers placement caused by contents of a hazardous nature that restrict the stowage choices available to the planner.

2.1.1 Introduction

The process of containerising cargo has brought about a revolution in marine cargo transport.[2] The container (a generic illustration of which can be seen in Figure 2-1) comes in many different forms and sizes. In general, all have a rectangular outer shape and a weather-proof outer shell that protects the cargo contained within.

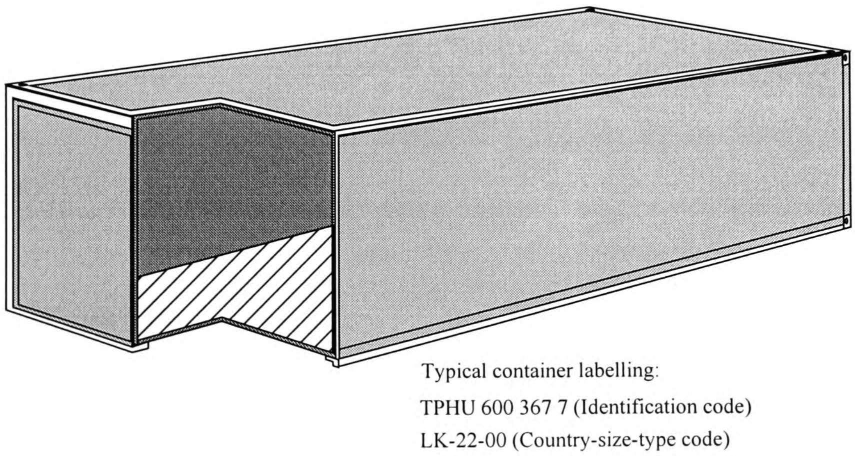


Figure 2-1 The Container

Each containers is assigned, on registration, a unique identifying code, and also three further codes associated with it that provide information about the container itself, namely:

- a country code;
- a size code;
- a type code.

The following explanation of the above codes illustrates the variety of containers in use.

2.1.2 The identifying code

Each container is labelled with its own identifier and a descriptive code that provides information about the containers place of origin, general dimensions and specific type.

2.1.2.1 The identifier

At registration each container is assigned a unique alphanumeric identifying code made up of a 4-letter owners code and a unique 7-digit container code.

2.1.2.2 Country of origin

The country code is simply a two or three letter abbreviation for the container's origin. A three-letter code indicates that the container was built before 1984, whereas a two-letter code indicates the container was built during or after this date. For example, a container labelled 'LK' indicates that it was registered in Sri Lanka after 1984.

2.1.2.3 Container Size

The size code consists of two digits which indicate the length, height of a container and whether or not it can accommodate a tunnel for a *gooseneck*. (This tunnel is a recess in the container floor designed to accommodate the so-called 'gooseneck' of a particular type of chassis or trailer, having a low container-carrying frame and a raised strut to allow connection to a standard tractor coupling.[3]) The first digit of the size code refers directly to the length of the container.

The following numbers correspond to the most common lengths in use:

(1) 10 feet (3000mm) long,

(2) 20 feet (6000mm) long,

(3) 30 feet (9000mm) long,

(4) 40 feet (12000mm) long,

(5) less than 10 feet long,

(6) between 10 and 20 feet,

(7) between 20 and 30 feet,

(8) between 30 and 40 feet,

(9) over 40 feet in length.

The International Standards Organisation (ISO) recommend that only the first four lengths are used. [3] The second digit of the size code indicates the height of the container and the presence of a tunnel:

(0) 8 feet high without a tunnel for a gooseneck,

(1) 8 feet high with a tunnel,

(2) 8 feet 6 inches high without a tunnel,

(3) 8 feet 6 inches high with a tunnel,

(4) more than 8 feet 6 inches high without a tunnel,

(5) more than 8 feet 6 inches high with a tunnel,

(6) 4 feet high without a tunnel,

(7) 4 feet high with a tunnel,

(8) 4 feet 3 inches high or more, with or without a tunnel,

(9) less than 4 feet high, with or without a tunnel.

Numbers (4) and (5) are commonly used to indicate so-called *high cubes*, containers that are 9’ 6” in height (an important development in containers, the implications of which are described in Appendix C), and respectively without, or with, a tunnel.

ISO recommend that a container should be 8’ wide, 8’, 8’ 6” or 9’6” high and either 10', 20', 30' or 40' in length with the 20' and 40' containers being the most common currently in use. [*Ibid.*] Containers can be stacked one on top of another with no additional support as long as they are of the same length and width. The containers that do not conform to the above recommendation are referred to as *non-standard* and have additional information associated with them indicating exact dimensions. It should be noted that some containers are actually frames that house a variety of different cargo types. Although the frame is of a standard size, the cargo can protrude thus affecting the actual dimensions of the 'container'.

2.1.2.4 Container type code

Containers have been classified by the ISO into nine general types, with numerous sub-types within each general type. The type code consists of two digits of which the first indicates the general type and the second indicates the sub-type. The majority of containers now fall into the following general classifications (although it should be noted that some operators still insist on using older types; 1988 revision of ISO codes [5]):

ISO Container Type Codes

First Digit	Container Type	Characteristic	Full Code
0	General purpose – openings at one or both ends	Basic type	00
		Plus full opening in one or both sides	01
		Plus partial opening in one or both sides	02
		Plus opening roof	03
		Plus opening roof and opening in one or both sides	04
1	Closed, passively vented	Vents less than 25 cm^2/m length	10
		Vents more than 25 cm^2/m length	11
		Non-mechanical	13
	Closed, ventilated	Mechanical, internal	15
		Mechanical, external	17
2	Dry-bulk, non-pressurised	Closed	20
		Vented	21

		Ventilated	22
		Airtight	23
		Livestock carrier	25
	Named types	Automobile carrier	26
3	Thermal	Refrigerated, expendable refrigerant	30
		Mechanically refrigerated	31
		Refrigerated and heated	32
		Heated	33
	Thermal, self-powered	Mechanically refrigerated	36
		Refrigerated and heated	37
		Heated	38
4	Thermal, refrigerated and/or heated, with removable equipment	Equipment external (low insulation value)	40
		Equipment internal	41
		Equipment external (high insulation value)	42
	Insulated	Insulated (low insulation value)	45
		Insulated (high insulation value)	46
5	Open-top, openings in one or both ends	Basic type	50
		Removable top member(s) in end frames	51 52
		Opening(s) in one or both sides	53
		Opening(s) in one of both sides and removable member(s) in end frame(s)	54
		Part opening in one side, full opening other	
6	Platform	Basic type	60
		Complete fixed ends	61
		Fixed free-standing posts	62
		Folding complete end structure	63
		Folding free-standing posts	64
	Platform, with	With roof	65

	complete superstructure	With open top	66
		With open top, open ends	67
7	Tank (test pressure)	For non-dangerous liquids (0.45 atmosphere)	70
		For non-dangerous liquids (1.5 atmosphere)	71
		For non-dangerous liquids (2.65 atmosphere)	72
		For dangerous liquids (1.5 atmosphere)	73
		For dangerous liquids (2.65 atmosphere)	74
		For dangerous liquids (4.0 atmosphere)	75
		For dangerous liquids (6.0 atmosphere)	76
		For dangerous liquids (10.5 atmosphere)	77
		For dangerous liquids (22.0 atmosphere)	78
8	Dry-bulk, non-pressurised, hopper type	Closed	80
		Vented	81
		Ventilated	82
		Airtight	83
	Dry-bulk, pressurised (test pressure)	Horizontal discharge (1.5 atmosphere)	85
		Horizontal discharge (2.65 atmosphere)	86
		Tipping discharge (1.5 atmosphere)	87
		Tipping discharge (2.65 atmosphere)	88
9	Air/surface	(Codes being developed)	90

For example, the basic type of container, the general purpose box with openings at one end or both ends, would be type coded as '00'. Not all the code numbers have yet been assigned, there being many spare numbers available for future expansion of the classified types.

2.1.3 Container content

This section explains the different types of cargo carried by container-ships, the variety of specialised containers and how special segregation consideration is given to special cargo.

2.1.3.1 General description

Content falls into three basic types:

- dry cargoes;
- liquid cargoes and bulk commodities;
- special cargoes requiring special handling.

Special cargo types require strict handling and are collectively referred to as '*specials*'. For example, a common requirement for particular types of special containers is the provision of a power supply to either cool or heat its contents.

2.1.3.2 Hazardous Cargo

Hazardous cargo, if not handled correctly, can put the ship, crew or other cargo at risk. Therefore, hazardous cargo is separated according to type and relationship with other types. The following hazardous cargo types are commonly carried:

i) Explosives

ii) Flammable gases

iii) Non-toxic, non-flammable gases

iv) Poisonous gases

v) Flammable liquids

vi) Flammable solids

vii) Spontaneously combustible substances

viii) Substances that are dangerous when wet

ix) Oxidising substances

x) Organic peroxides

xi) Poisons

xii) Infectious substances

xiii) Radioactive material

xiv) Corrosives

xv) Other miscellaneous dangerous substances

2.1.3.3 Segregation of hazardous cargo

The properties of substances within each class of hazardous cargo may vary greatly. The IMDG publication [5] documents particular *segregation* requirements (i.e. rules concerning distances between hazardous cargoes). Since the properties of substances or articles within each class may vary greatly, the individual schedules would always be consulted for particular requirements for segregation; these schedules take precedence over the general segregation requirements. However, Table 2-1 shows the general requirements for segregation between the various classes of dangerous goods. Each of the hazardous cargo types described above is given an IMDG code (*e.g.* Organic Peroxides is given the code 5.2). Cross-referencing a hazardous cargo type with another on Table 2-1 either gives a number from 1 to 4 or the letter X. Segregation class 1 requires that the cargo be stowed away from each other. Segregation class 2 means that they must be separated from each other. Segregation class 3 means that they must be separated by a complete compartment or hold. Segregation class 4 means that they must be separated longitudinally by an intervening complete compartment. Where the segregation class is labelled with an X, this means that more specific segregation may be required which, if any, may be found in the individual schedules.[5]

For example cross-referencing flammable solids, 4.1, with spontaneously combustible substances, 4.2, gives the number 1, which means that these cargo types must be stowed *away* from each-other. For specific information about the segregation of different types of the cargo refer to the Stowage and Segregation Guide to IMDG. [*Ibid.*]

Class	1.1 1.2 1.5	1.3	1.4	2.1	2.2	2.3	3.0	4.1	4.2	4.3	5.1	5.2	6.1	6.2	7.0	8.0
Explosives 1.1,1.2,1.5	X	X	X	4	2	2	4	4	4	4	4	4	2	4	2	4
Explosives 1.3	X	X	X	4	2	2	4	3	3	4	4	4	2	4	2	2
Explosives 1.4	X	X	X	2	1	1	2	2	2	2	2	2	X	4	2	2
Flammable gases 2.1	4	4	2	X	X	X	2	1	2	X	2	2	X	4	2	1
Non-toxic, non-flammable gases 2.2	2	2	1	X	X	X	1	X	1	X	2	1	X	2	1	X
Poisonous gases 2.3	2	2	1	X	X	X	2	X	2	X	X	2	X	2	1	X
Flammable liquids 3.0	4	4	2	2	1	2	X	X	2	1	2	2	X	4	2	X
Flammable solids 4.1	4	3	2	1	X	X	X	X	1	X	1	2	X	4	2	1
Spontaneously combustible substances 4.2	4	3	2	2	1	2	2	1	X	1	2	2	1	4	2	1
Substances that are dangerous when wet 4.3	4	4	2	X	X	X	1	X	1	X	2	2	X	2	2	1
Oxidising substances 5.1	4	4	2	2	X	X	2	1	2	2	X	2	1	4	1	2
Organic peroxides 5.2	4	4	2	2	1	2	2	2	2	2	2	X	1	4	2	2
Poisons 6.1	2	2	X	X	X	X	X	X	1	X	1	1	X	1	X	X
Infectious substances 6.2	4	4	4	4	2	2	4	4	4	2	4	4	1	X	4	4
Radioactive materials 7.0	2	2	2	2	1	1	2	2	2	2	1	2	X	4	X	2
Corrosives 8.0	4	2	2	1	X	X	X	1	1	2	2	X	4	2	X	X

Table 2-1 Segregation Table

2.2 The Container-ship

This section describes the different types of ships used to transport containerised cargo.

2.2.1 Types of Container-ships

Container-ships come in a variety of designs. These may be classified as follows [12]:

- Roll On Roll Off (RoRo) ships;
- Lift On Lift Off (LoLo) ships that can be divided into:
  - ♦ barge carriers;
  - ♦ conventional, semi-container-ships;
  - ♦ the cellular container-carrier.
- combination types (which are combinations of the above types).

RoRo (Roll on/Roll off) vessels are loaded by having vehicles drive into their cargo areas whereas *LoLo* (Lift on/Lift off) are ships that have their cargo loaded and unloaded from above by cranes. Barge carriers usually transport cargo on short hauls and are relatively small. Semi-container-ships are usually older vessels that have been altered to allow containers to be carried.

Combination type vessels combine aspects of other types to create a wide range. Lastly, cellular container-carriers, the primary interest of this study, are designed specifically for the transportation of containers. The title for this class of ship comes from the array-like organisation of the cargo-space made up of cell-guides and slots (described in Section 2.3). All container-carrying ships come in different shapes and sizes and are described according to their Twenty-Foot Equivalent Unit (TEU) capacity. An example of such a vessel is illustrated in Figure 2-2 (the terms *bay* and *hatch* are explained in Section 2.2.2). Below deck cargo holds are covered by hatch-lids that both support above-deck cargo and enclose the hold, protecting it from the environment.

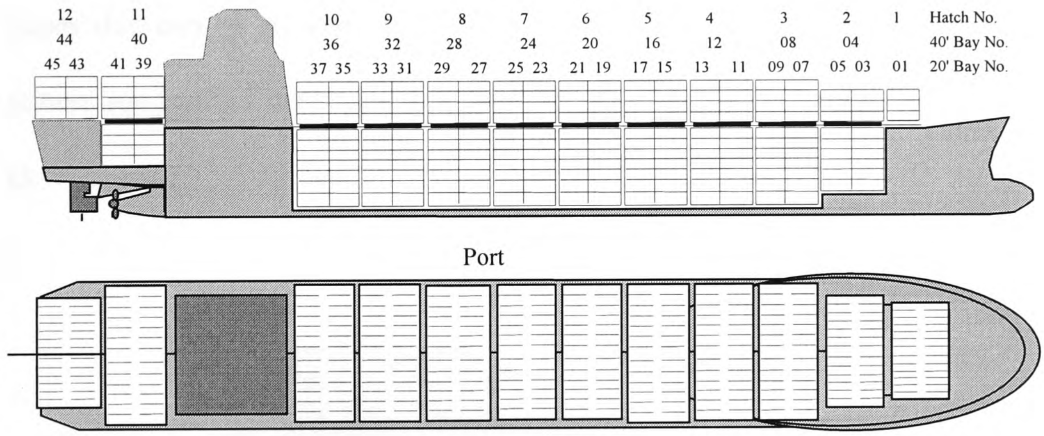


Figure 2-2 The Cellular Container-ship

2.2.2 Container-ship cargo space geometry

Container *cells* (spaces allocated for holding containers, explained in section 2.3.1) are grouped into *stacks* (the name given to a vertical grouping of containers) and then *bays* (a collection of stacks across the width of the ship, see Figure 2-3). Each layer of these container spaces is called a *tier* (a horizontal group of cells within a bay). A bay may only be *on-deck* (literally on the deck exposed to the elements) or it can include a cargo-space *below-deck* (below the deck within the ship, enclosed by a hatch-lid). Bays are grouped together and are associated with a *hatch* number. A hatch is a collection of bays, usually between one and three in number running laterally through a ship. Each hatch number generally corresponds to the points of access to under-deck cargo along the ship numbered from the front to the rear. Under-deck bays that correspond to a hatch are separated from each other by *bulkheads* (walls, that when combined with the outer hull of the ship creates enclosed *compartments* or *holds*). *Cell-guides* facilitate the lowering of containers to their proper positions below-deck and resist the horizontal movement of containers caused by the movement of the ship during a voyage. The cell-guides determine the size of container that can be stowed under-deck. Since there are no cell-guides on deck, containers are instead constrained by *lashing* (the tying down of containers to the deck).

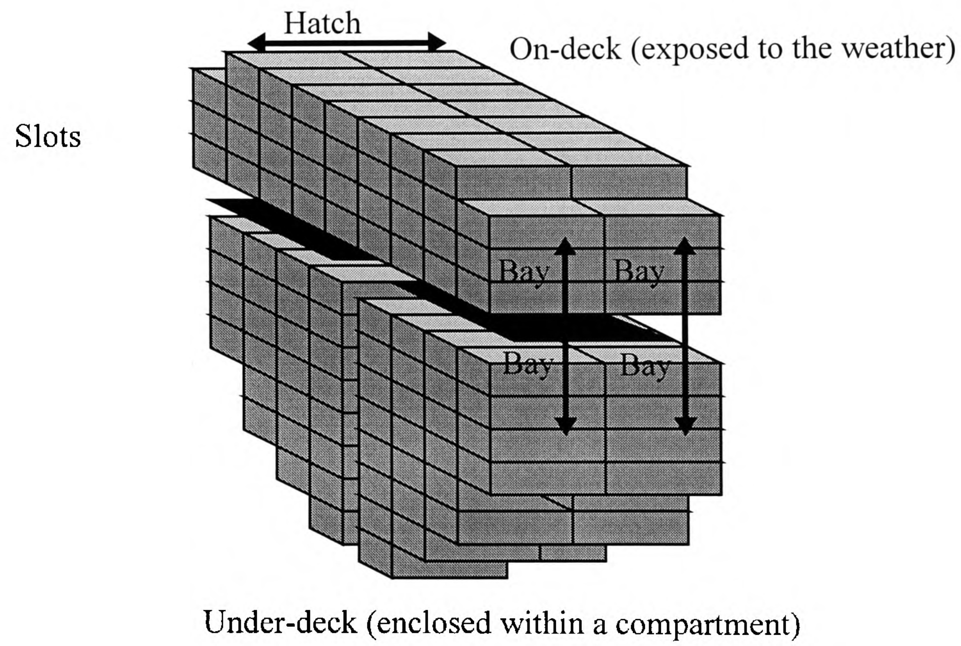


Figure 2-3 Typical Bay Configuration

Cellular container-ships, that fall into the above general description, usually have a uniform orientation of their cargo-space where the position of cargo is identified by an address (explained in Section 2.3). However, in some instances ships can not store containers according to this uniform system. [6,12] Most notably, containers are stowed on some vessels at different orientations to the ship's longitudinal axis. [6]

To accommodate container-ships that do not fall into the uniform pattern described above, the specific geometric co-ordinates of each container are used to identify its position within the cargo-space. The co-ordinates used to identify the position of each container are the *Longitudinal Location* (LL - distance along the longitudinal axis of the vessel), the *Transverse or Horizontal Location* (TL - distance along the horizontal axis of the vessel), the *Height* above the *Base Line* (height above the base of that stack) and *Rotation* (angle of the stack in relation to the longitudinal axis) that provide the precise co-ordinates of each container stowed within the cargo space.[6] Therefore, the LL, TL, Height and Rotation are important since they give the co-ordinates of a specific container (see Figure 2-4). [Ibid.]

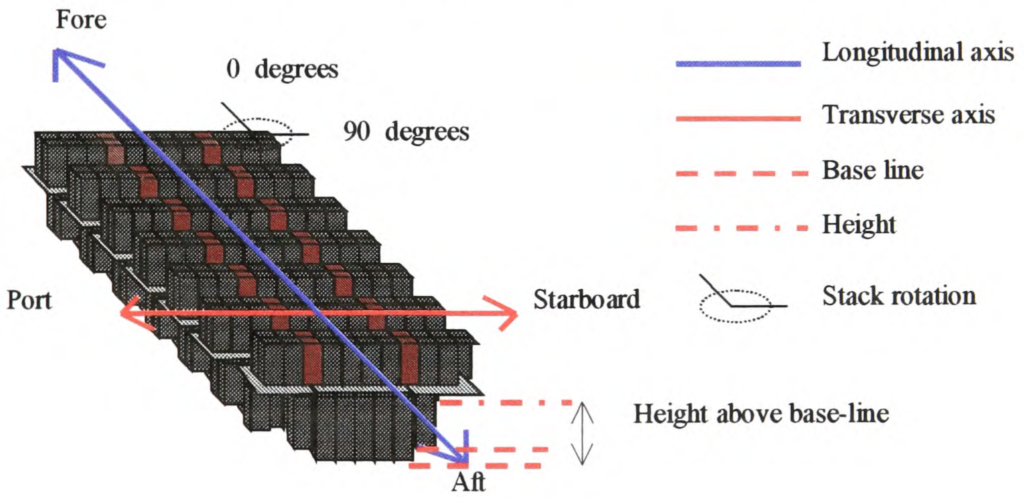


Figure 2-4 Cargo Space Geometry

As explained above, cargo can be placed on deck exposed to the environment, or under deck in enclosed spaces called holds. The possible height of a stack of containers can vary depending upon the ship's design (to a maximum stack height of nine below deck and six above deck). Containers are loaded from the bottom to the top of a ship, tier by tier. This tier by tier ordering of loading creates the requirement that those containers for earlier ports of discharge are above those for later ports of call. A container that violates this condition is referred to as *over-stowage* (described in Section 3.2.2).

Under-deck bays are usually dedicated to a specific container size, either 40 feet or 20 feet in length. The same is not true of on-deck bays, most of which allow mixes of different container sizes. However, in practice, containers are stacked according to length in order that technical problems associated with loading and securing together containers of different sizes be avoided. Containers are stowed on the ship in vertical stacks above and below deck, with large hatch-lids separating above and below deck stacks. To remove the hatch-lid and permit access to under-deck containers all containers stowed on the hatch-lid must first be removed.

2.3 Container address system

2.3.1 The cell

The *container address system* [3] is the term used to describe the labelling of standard container stowage locations within the cellular ship. In such a vessel, containers are stacked fore and aft, above and below deck, in cells. Each cell is considered to be 20’ long, 8’ wide and 4’3” high. Note that each container position, or *slot*, is shown on documents as 8’6” high, so each rectangle actually represents two cells, one above the other. Many positions show a 40’ long stowage location, too, in which case each slot represents four of the basic cells. The full address of a container is made up of a *longitudinal*, *transverse* and *vertical* reference that are explained in detail below.

2.3.2 Longitudinal labelling of cells

The bays are numbered from fore to aft, using a two-digit numbering system (for example, the first three bays in Figure 2-2 are shown as 01, 03 and 05). The system is complicated by the need to distinguish between two types of bay: those suitable only for 20’ or 40’ length containers, and those which accommodate 20’ and 40’ length containers. The system gets around this by giving odd numbers (05, 07 *etc.*) to bays for 20’ containers and even numbers (02, 06 *etc.*) to bays with 40’ positions; in Figure 2-2 the first bay is labelled 01 to indicate a 20’ restriction on length of containers whereas the next two bays can accommodate either two 20’ containers, giving reference numbers 03 and 05, or one 40’ container with a reference of 04.

The length of container that can be accommodated in an under-deck bay of a given container-ship depends upon the cell guides fitted. Below-deck bays are usually built for either exclusively 20' or 40' containers, but most above deck bays can be loaded with either size of container. A vertical dotted line dividing a compartment on the *General Arrangement Plan* (a plan showing a longitudinal cross-section of a container-ship, see Section 3.3.2.1 for a full description) indicates that either 40' or 20' containers can be stowed there. In these cases, the bay number assigned to a container within the bay will depend on the size of the container (*e.g.* 02 to a 40' container, and either 01 or 03 for a 20' container).

2.3.3 Transverse labelling of cells

Within the container address system, each vertical stack is assigned a number; when combined with the bay number this uniquely identifies each stack. A few ship operators number the stacks from left to right (or from right to left). However, most operators now follow the convention of numbering from the *centre line* of the vessel (the line running longitudinally along the centre of the ship), using two digit numbers. Stacks to starboard are given odd numbers (e.g. 01, 03), those to port are given even numbers (e.g. 02, 04). Where there is a vertical stack on the centre line, it is given the number 00. Examples of this system of stack numbering can be seen in Figure 2-5. The stack numbers are usually painted on the hatch *coamings* (a raised frame round a ship's hatchway for keeping out water), at the top of the cell guides, to assist in the identification of the required slot for ship loading and discharging. This is extremely helpful to the crane operator and to those supervising the ship operation. If the markings were not displayed, the stacks would have to be counted each time a container is lifted or stowed.

2.3.4 Vertical labelling of cells

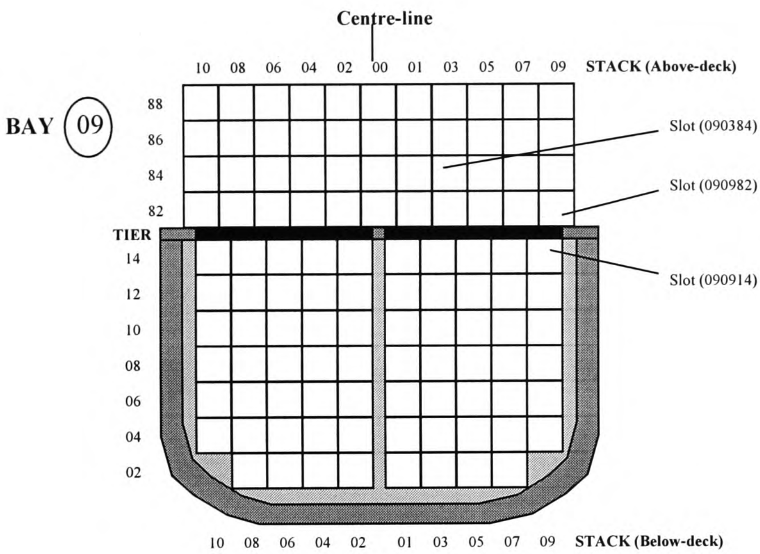


Figure 2-5 Container-ship cross-section showing container address system

A vertical stack of cells is made up of a series of levels called tiers. The container address system is completed by adding a two-digit number to indicate in which tier the container is stowed, with all containers on the same level within their respective stacks having the same tier number. The number is allocated according to container height. The most common full height 8'6" containers are allocated even tier numbers whereas the half height 4'3" containers are allocated odd tier numbers. It is the full height (even) numbers that are printed on the bay profiles in the outline plan, and the drawn rectangles in Figure 2-4 (and in standard documentation) represent this full height container.

Note that the bottom tier 01/02 refers to the lowest tier in that hatch. Because of the shape of the hull and the presence of tanks and other non-cargo areas, below-deck stacks away from the centre line to port or starboard may well start at a higher level than those of the centre line, but the tier always carries the number of the slot at the same level as the centre line (see Figure 2-4). The above-deck tiers are numbered in a similar way, but here the initial digit of the each tier is 8, not 0. Therefore, the lowest tier of containers carried on the hatch cover or *weather deck* (the deck of the ship that is exposed to the elements) is numbered 81/82, that above it 83/84, and so on. Considering all three components of the full container address system, the numbering of slots is a straightforward system based on simple principles and allows containers to be located. The three components of the full container address system come together to form the complete *cell address*. The cell address consists of three pairs of digits, indicating in turn the bay number, stack number and tier number. In Figure 2-4, the address 090914 indicates the cell found below deck in bay 09; stack 09; tier 14 (seven high in the stack). Similarly, the address 090384 indicates the cell found above-deck in bay 09; stack 03; tier 84 (two high in the stack).

2.3.5 Intact stability

Part of the planners task is to ensure that the vessel will remain in a stable condition. Since ship stability theory is a large subject, the purpose of this summary is only to cover theory related to container-ship stowage planning. For a more thorough discussion of intact stability refer to 'Principles of Naval Architecture' (1980). [7] After an initial introduction to the general principles of intact stability, the following sections describe the *angle of heel* (the transverse, horizontal angle of the deck in relation to the waterline), the *GM* (the ship's metacentric height) and the *angle of trim* (the longitudinal, horizontal angle of the deck in relation to the waterline).

2.3.5.1 General Principles

When a solid body is immersed in water, it experiences an upward thrust equal to the weight of the water it displaces. [7] This weight is referred to as the *displacement* (the weight or volume displaced by a body in liquid) of the vessel. The displacement of a vessel is the sum of its light condition weight and the total weight of the cargo and stores carried. Any temporary changes, such as variations in cargo, ballast or fuel, to the light condition of the vessel will also be included. This total weight represents the effect of gravity. The position of the waterline on the hull of the vessel is determined by the displacement. In this state the sum of all forces acting upon the ship is zero, and the ship is referred to as being in *equilibrium*.

If more weight is placed to *port* (the left side of a ship when facing its front, or *bow*) or to *starboard* (the right side of a ship when facing its bow), or vice versa, then it will 'lean' in that direction. This leaning is termed *heeling*. (Heeling is the term used to describe a vessel that is inclining from the vertical, either to port or to starboard, and is measured in degrees, for example a ship may be heeling 3 degrees to port.) Because having a ship heeling to one side is not an ideal state for the vessel to be in, cargo weight must be spread as evenly as possible from port to starboard so that heeling can be prevented, or at least brought within an acceptable level of tolerance. Filling *ballast tanks* (large containers or reservoirs for liquids, in this case sea-water, used to stabilise a vessel) with *ballast* will counteract heeling. Use of ballast should be kept to a minimum since the ballast is effectively additional 'cargo'.

2.3.5.2 Upsetting and Righting Moments

For intact stability calculations gravity is treated as a single force acting vertically downward through the ship's *centre of gravity* (G - the point through which the single vector that is the vector sum of two or more others vectors of the gravitational forces on a body always acts). The upward force of *buoyancy* (the tendency of a liquid to keep a body afloat) acts through the *centre of buoyancy* (B - the point through which the single vector that is the vector sum of two or more others vectors of the buoyancy forces on a body always acts), located at the *geometric centre* of the ship's *underwater body*. [Ibid.] When the ship is in equilibrium, G and B lie on the same vertical line. As a vessel takes on more weight, the *water-line* (a line marking the level reached by a body of water), or *draft*, rises and the centre of buoyancy rises in relation to the keel (K - one of the main longitudinal structural members of a vessel, running along the bottom of the hull, to which the frames are fastened.). Various values for KB are calculated and form part of the *hydrostatic* data (statistics used by Hydrostatics, a branch of science concerned with the mechanical properties and behaviour of fluids that are not in motion) for the ship.

When a disturbing force, such as the weight distribution of cargo, acts upon the ship, the part of the hull of the ship which is underwater changes bringing about a relocation of the centre of buoyancy (B). The vessel is no longer in equilibrium since the forces acting at B and G are not equal and opposite.

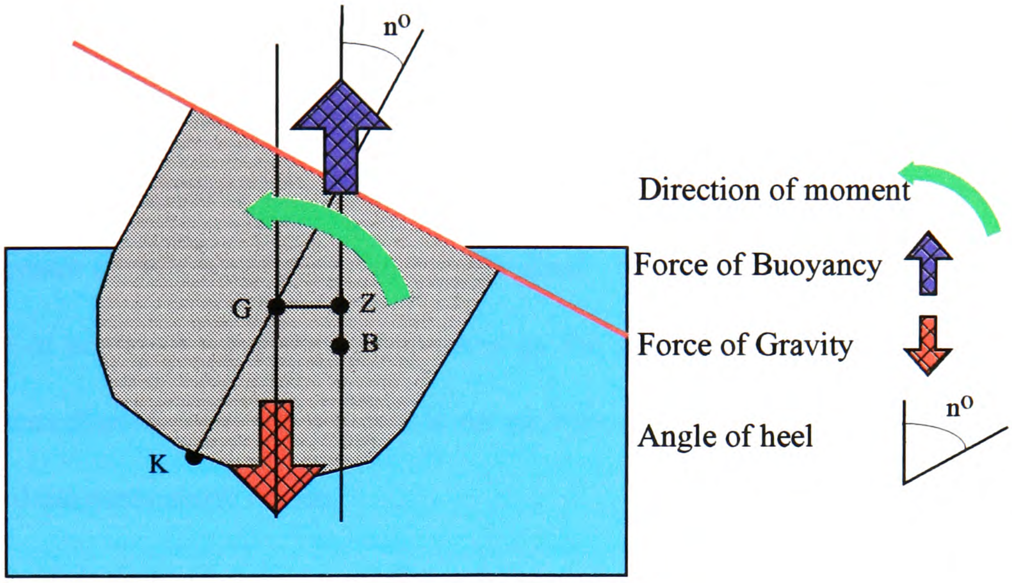


Figure 2-6 Righting Moment

The newly formed couple of forces acting at B and G produce either a *righting moment* (a tendency to rotate the ship back to equilibrium - see Figure 2-6), or an *upsetting moment* (a tendency to rotate the ship away from equilibrium - see Figure 2-7), dependant upon their relative positions.

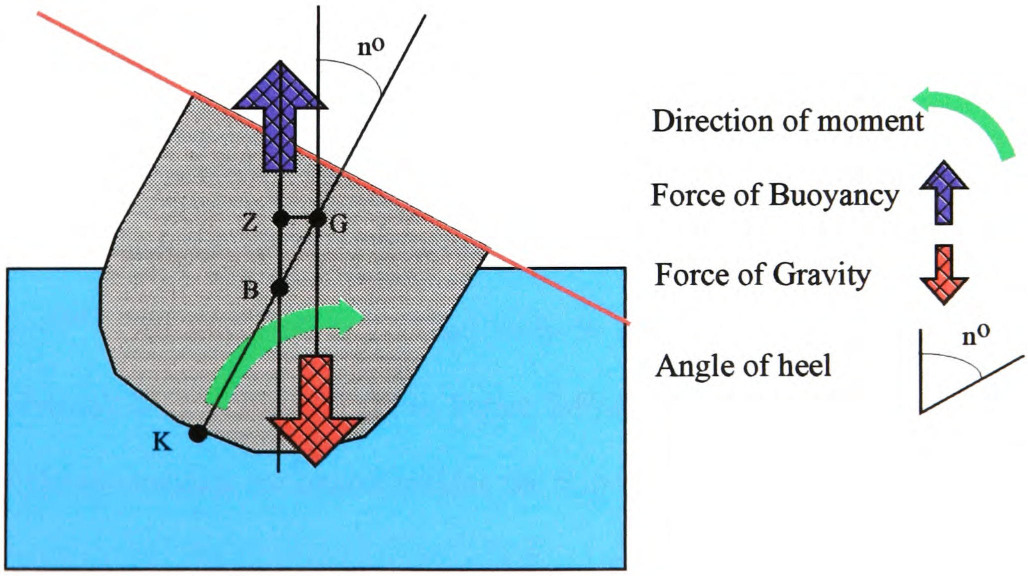


Figure 2-7 Upsetting Moment

2.3.5.3 Transverse Metacentre

The *Transverse Metacentre* of a ship, M, is the intersection of a vertical line through the centre of buoyancy of a floating body at equilibrium with a vertical line through the centre of buoyancy when the body is tilted. M is the intersection point of two lines of action of the force of buoyancy as the ship heels (see Figure 2-8). The distance from B to M when a ship is on an *even keel* (well balanced and steady) is called the *metacentric radius*.

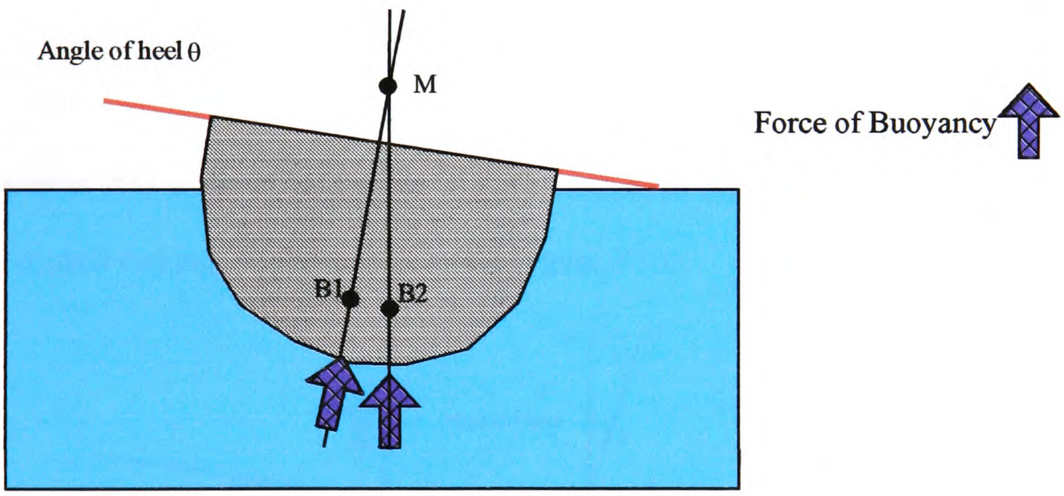


Figure 2-8 Location of Metacentre

The distance from G to M (see Figure 2-9) is known as the ship's *metacentric height* (GM). The metacentric height is very important in determining the ship's *righting ability* [ibid.] (the ability of a disturbed body in a liquid to bring itself back into equilibrium). If M is above G (as in Figure 2-9) then the GM is said to be positive and righting moments are created making the ship stable.

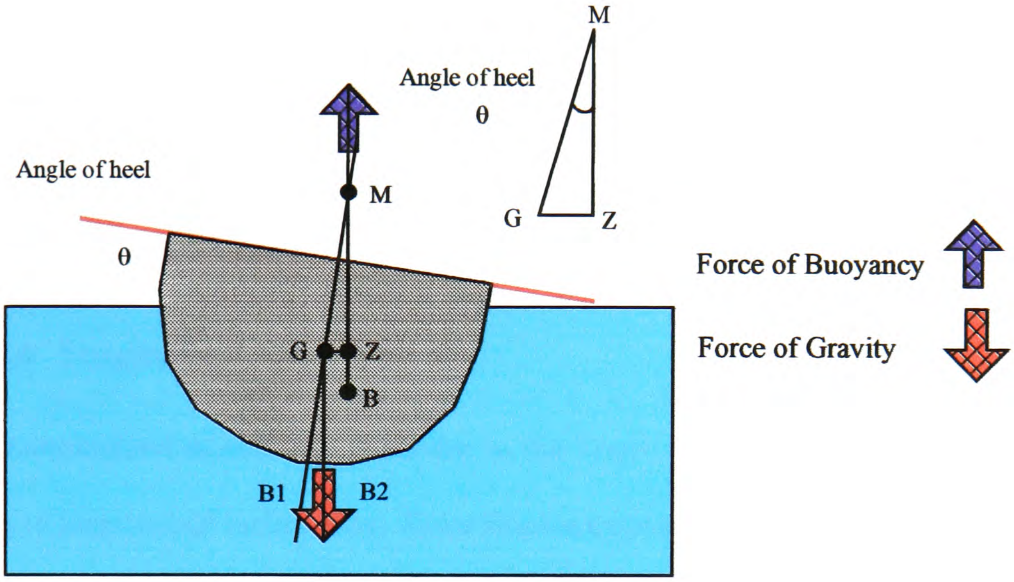


Figure 2-9 Stable Condition

However, if G is above M then the GM is said to be negative and upsetting moments are created making the ship unstable (see Figure 2-10).

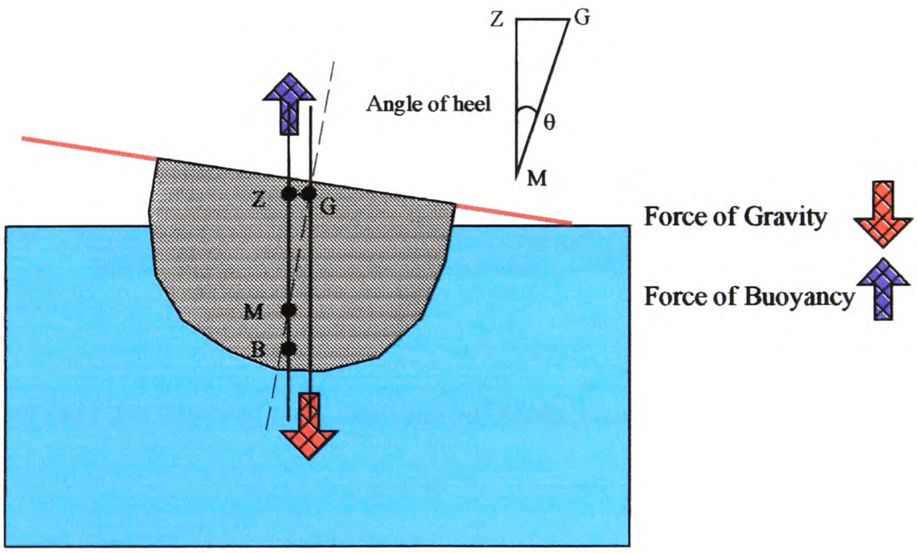


Figure 2-10 Unstable Condition

When the GM is large then the righting moment is large, reducing the angle of heel. A ship with a large GM resists rolling and is said to be *stiff* and, conversely, when the

GM is small a ship will roll slowly and is said to be *tender*. A large GM is preferable for resisting flooding if a ship is holed. However, a small GM is preferable since this will permit the ship to ride waves more easily. A range of acceptable values is normally associated with a ship.

2.3.5.4 Longitudinal Metacentre

The *Longitudinal Metacentre* is similar to the Transverse Metacentre except that it refers to longitudinal inclinations. Since ships are not symmetrical forward to aft, the centre of buoyancy at various even-keel waterlines does not always lie in a fixed transverse plane, but may move forward and aft with changes in draft. The upward force of buoyancy acts through the centre of buoyancy (B), which, as has been stated, will vary with different drafts.

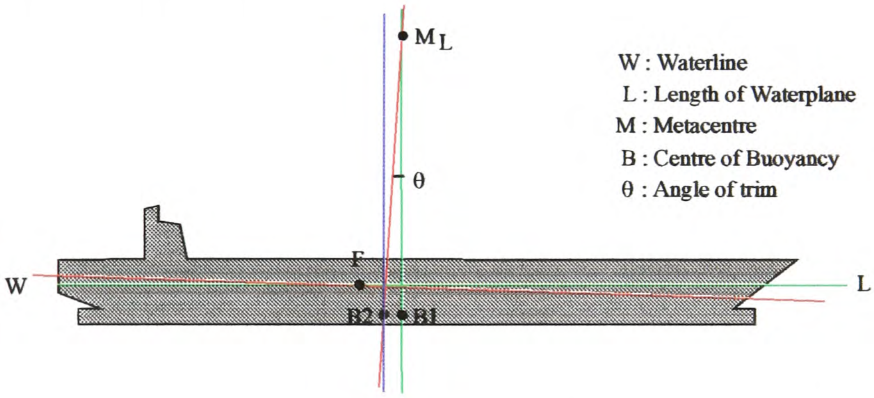


Figure 2-11 Location of the Longitudinal Metacentre and Metacentric Height

When a disturbing force acts upon the ship the shape of the ship's underwater body changes bringing about a relocation of B. The vessel is no longer in equilibrium since forces acting upon B and G are not equal and opposite. The newly formed couple of forces through B and G produce a change in the ship's trim, either to forward or to aft (see Figure 2-11). The vessel will incline about an axis through the geometric centre of the waterline. This point is known as the *centre of flotation.*

Trim (T) is defined as the difference in draft forward (D_f) and draft aft (D_a). Trim is either by the bow or by the stern. Locating the Longitudinal Metacentre is similar to locating the Transverse Metacentre. Since a ship is not symmetrical fore to aft the under-water shape of the vessel will be different to the shape above the water-line. However, to maintain the same displacement they must have the same volume.

The Longitudinal Metacentre is normally far above the centre of gravity, therefore the longitudinal metacentric height is almost always positive. The longitudinal metacentric height is a measure of the ship's resistance to trim. The longitudinal metacentric height is normally positive, thus making the ship inclined to right any change in trim. Trim is measured in degrees either to *fore* (to the front of the vessel) or to *aft* (to the rear of the vessel) and reflects the angle of the ship in the water. For example, a ship level in the water would have a zero trim. A small trim by the stern is preferable.

2.3.6 Adverse Structural Moments

This section describes the forces that can distort the physical structure of a ship.

2.3.6.1 Stress

The discussion on stability, thus far, has assumed an even distribution of cargo weight. As the weight of cargo is rarely distributed evenly across a ship, the ship is put under varying levels of stress at different points along its structure. Adverse stress to the structure of the ship can occur longitudinally and transversely. Longitudinal stress, caused by an uneven weight distribution from bow to stern, will occur along the length of the ship and is often called *bending.* [7] Transverse stress, caused by an imbalance in weight distribution from port to starboard, will also occur along the length of the ship and is often referred to as *torsion.* [Ibid.]

2.3.6.2 Bending and Torsion

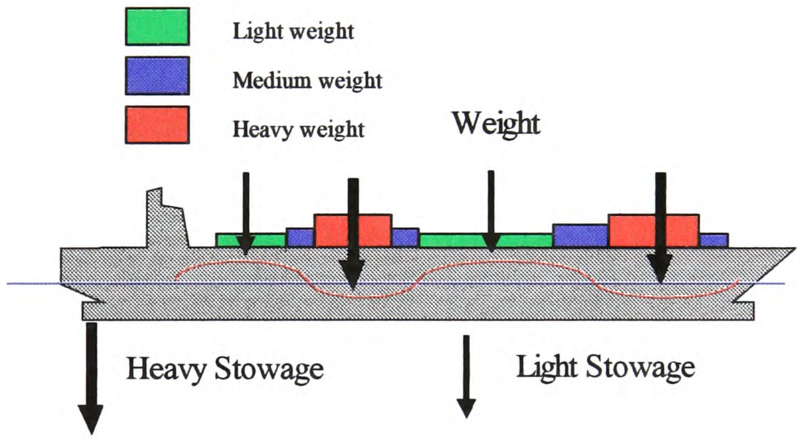


Figure 2-12 Bending Moments

A ship will react, and bend, in much the same way as a metal bar when pressure is applied upon a point along its length. For example, an uneven distribution of weight along the length of the ship will cause the structure to bend vertically in proportion to the weight exerted. This bending can be seen in Figure 2-12, in which the, exaggerated curved line illustrates the shape of the ship, fore to aft, brought about by the varying weights along the vessel's length. Therefore, the weight of cargo must be spread as evenly as possible along the length of the ship.

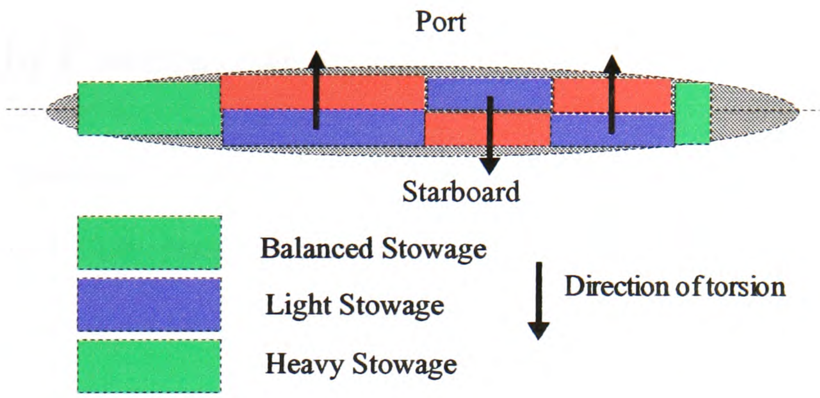


Figure 2-13 Torsion

Torsion is caused by an uneven weight distribution transversely at points along its length (see Figure 2-13). Given that a total distribution of weight to starboard of the centre-line will cause the vessel to heel in that direction, a similar effect is noted when viewing cross sections of the ship at various points along its length. In affect, the structure of the vessel will twist in the direction dictated by the distribution of weight. Therefore, torsion can be alleviated, primarily, by an even distribution of weight across bays and, secondarily, by a judicious use of ballast.

2.4 The Container-terminal

To understand all factors that affect shipping operators efficiency it is necessary to understand the processes that occur at a container-terminal. This section outlines the following areas as an introduction to considerations that impinge upon stowage planning:

- container terminal organisation;
- container processing within the terminal;
- how terminal efficiency is measured;
- how information is passed between interested parties.

2.4.1 Organisation

A container-terminal is a primary node where transhipment of containers occurs between land transport and sea transport, as seen Figure 2-14. [2] Container-terminals form an essential link in the transport chain of a container.

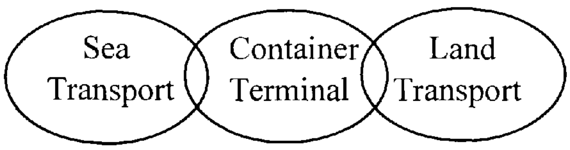


Figure 2-14 Container-terminal as a transportation link

A container-terminal fulfils two functions: the *transhipment* of containers (transfer from one cargo carrying vessel or vehicle to another [75]) and the temporary storage of containers. [8] The key features of the container terminal that affect the stowage planning task are:

- *Berth* (a place assigned to a ship at a mooring [75]) lengths, since cranes may be unable to manipulate cargo if the berth is shorter than the container-ship;
- Number of ship-to-shore gantry cranes at the designated berth, since stowage will be planned to maximise crane usage;
- Equipment requirements and usage, since there may be specific transhipment limitations associated with a berth or terminal that influence stowage planning, such as a maximum permissible height of an on-deck stack.

The layout of a container-terminal depends on the specific cargo handling equipment used. However, most ports usually contain the following components (illustrated in Figure 2-15), the function of which are explained below:

- The *dock area* includes the ship berths and the waterfront area where cranes operate on fixed rail tracks (illustrated in Figure 2-16);
- The *container storage area* (container yard) is where *inbound* (cargo unloaded at the terminal) and *outbound* (cargo ready to be loaded onto container-ships) containers are temporarily stored before being moved to their next destination;
- The *gate processing area* is where incoming and outgoing containers are processed. Containers are weighed, where necessary, and then allocated yard storage slots;
- Container consolidation and unpacking activities are carried out at the *container freight station*;

- Load planning, yard planning, information processing and control functions are performed within the *administration area*;
- Container terminal maintenance is co-ordinated within the *maintenance area*.

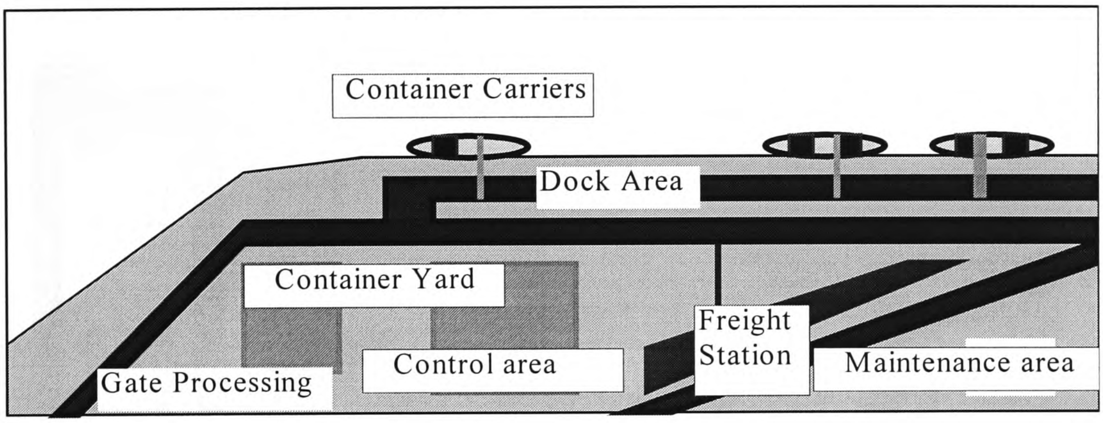


Figure 2-15 A Container Terminal

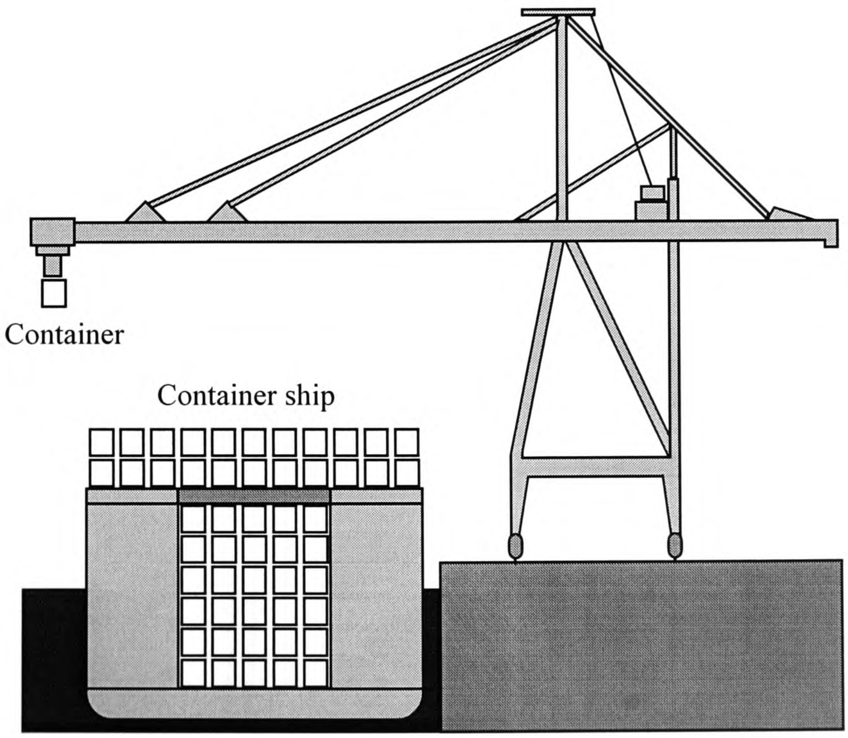


Figure 2-16 Ship berth with crane

The organisation of the container-yard is dictated by the type of vehicles used to transport and store containers around the yard, and to and from the ship during loading and unloading by cranes. Some of these vehicles are illustrated in Figure 2-17.

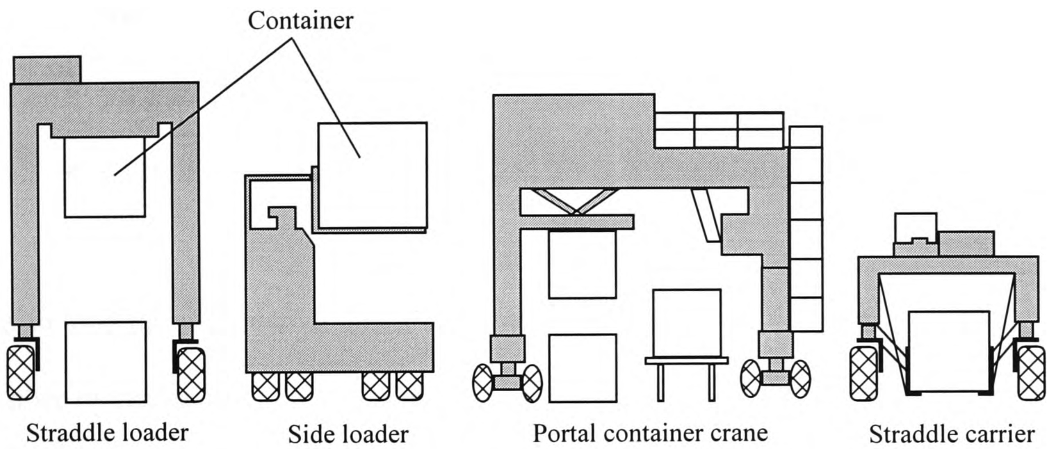


Figure 2-17 Specialised container lifting equipment

A typical method of organising the yard involves the use of a *Transtainer* (a type of straddle loader used to process containers within the container-yard, see Figure 2-18), transport vehicles of some sort (usually a truck) and a crane.

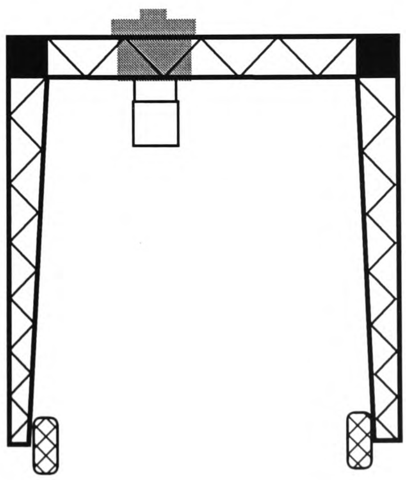


Figure 2-18 A Transtainer

The dockside loading sequence follows the following format:

- an export container is received;
- the export container is stowed in a pre-determined storage location in the container-yard;
- a Transtainer removes the container from the yard and delivers it to a yard truck;
- the yard truck proceeds to the crane area;
- a crane collects the container and loads it aboard the ship.

The unloading sequence follows the reverse of the above loading sequence. The number of cranes that are available for loading and unloading varies from container-terminal to container-terminal and from berth to berth but is usually in the range of one to three. An illustration of ship loading and unloading with two cranes is shown in Figure 2-19. It will be seen in subsequent chapters, the number of cranes working simultaneously on a ship is an important planning consideration.

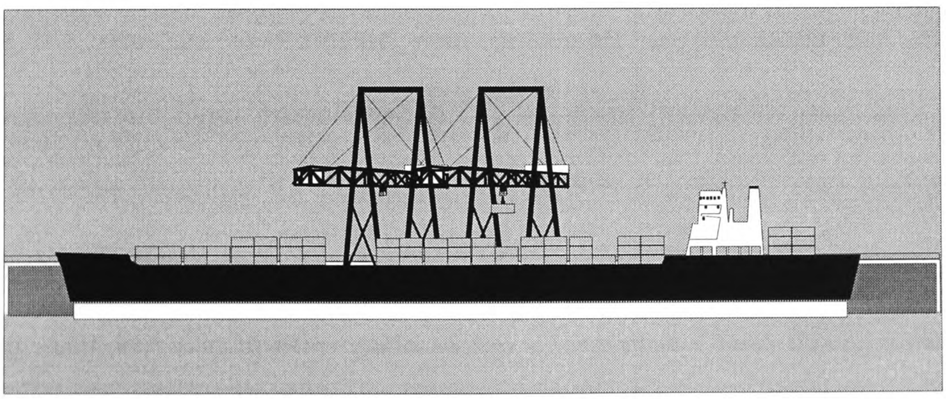


Figure 2-19 Container berth with two cranes in operation

The yard is organised into blocks with provision for trucks and transtainers to move containers. Though it is not always possible, most ports try to segregate containers by port of destination, length and weight and by type. If it is necessary to move the Transtainer to another part of the yard, the vehicle can move slowly with the Transtainer's wheels being rotated through ninety degrees to permit perpendicular movement.

2.4.2 Port Efficiency

Keeping the distance each container travels to a minimum whilst in addition to keeping the number of times each containers is handled to a minimum results in lower material-handling costs. Therefore, the efficiency of a container terminal is a measure of the distance travelled by containers and the number of movements of containers made. [4] Large savings in direct (material-handling) and indirect (administration) costs have been made at some container-terminals by computerising the internal processes of container-terminals. [2, 4, 8, 9, 10, 11]

2.4.3 Information flow

Linking the stowage co-ordinator with marketing organisations and terminals through *Electronic Data Interchange* (EDI) has been proven to save time and cut costs. Stowage planning for containerised cargo is a complex task that has seen considerable improvements by the use of computerised tools, not only in the planning stage but also in other parts of the information chain, through the use of EDI, such as in in-coming and out-going data and the statistical follow up after a voyage.

Improving the information flow between the stowage planner, the *stevedore* (the person employed to load and unload ships) and the ship's cargo officers, as well as sources of bookings and forecasts would improve efficiency.

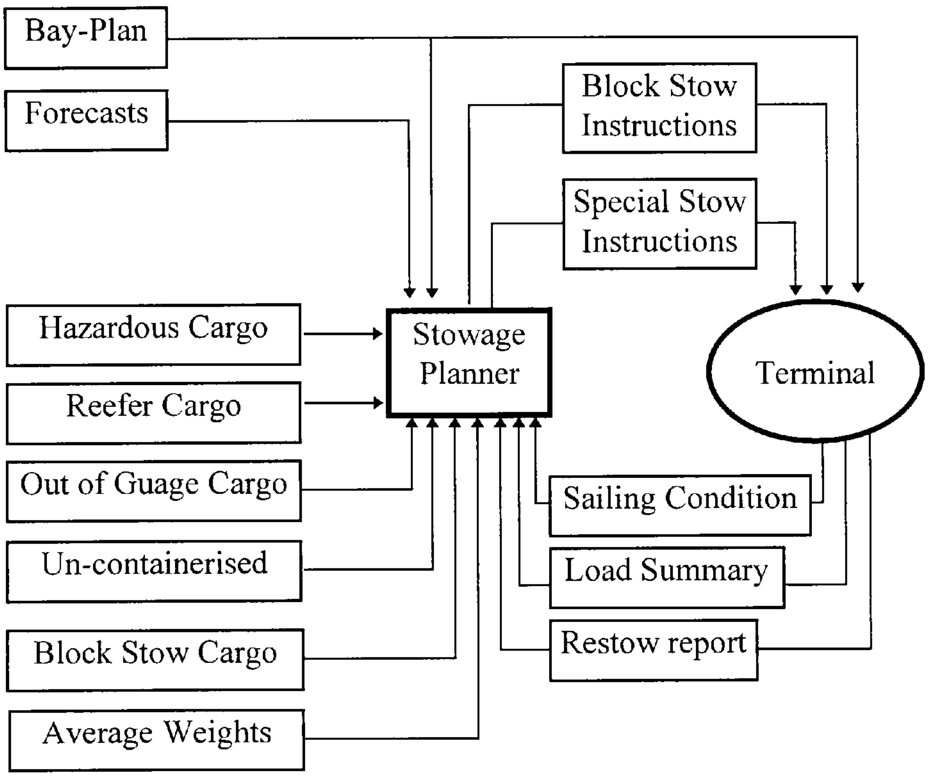


Figure 2-20 Information flow

The information flow to the stowage planner is used to prepare a *profile* (a colour paper bay-plan, described in detail in Section 3.3.2). The profile is regarded as being the planner's single most important tool. [9]

3 THE STOWAGE PLANNING PROCESS

3.1 Introduction

This chapter presents a description of the stowage planning process and of the individuals concerned with performing the operation of planning. This chapter goes on to identify aspects of, and constraints on, the planning process that must be taken into account by an automated system.

3.2 Problem Description

In this section, the important stowage planning terms are explained, the concept of the container re-handle is outlined, and the individual entrusted with the planning of container-ship stowage is identified.

3.2.1 Terms and definitions

Since terms and associated definitions vary from country to country and indeed from ship operator to ship operator, a brief description of how they are used within this document follows:

- The *strategic pre-planning* (long term planning of container stowage) and *tactical planning* (short-term planning of container stowage) phases of containerised cargo transportation between container-terminals around the world govern the stowage of cargo on deep-sea container-ships.
- The *pre-planner* is the individual concerned with planning the stowage of cargo during the strategic planning phase, so that the long-term efficiency of cargo stowage is as close to optimal as possible.

- The stowage *pre-plan* generated during the strategic pre-planning phase incorporates a large amount of forecast information. Therefore a stowage pre-plan is a generalised stowage model, that will become more specialised as more accurate information on actual cargo replaces forecasts. The pre-planning stage employs an abstract view of the container-ship. That is, the container-ship is divided up into logical areas where containers will be placed.
- The *planner* is the individual concerned with making allocations of cargo, during the tactical planning phase, to stowage locations on the container-ship.

Both the ship-operator and the terminal owner employ planners, the main distinctions between the two being that:

- the ship-operator planner must consider the long term effect stowage decisions have upon the efficiency of the vessel at terminals further along in the rotation;
- the terminal planner is concerned with making specific cargo placements of cargo, taking into account the short term implications of these placements, usually based upon the plans provided by the ship-operator's planner, such that all constraints are met and terminal efficiency is maximised;
- the plans prepared by the ship-operator planner are often generalised, facilitating the terminal planner's task, although specific container placements are often made (the degree of flexibility offered to the terminal varies from operator to operator, based upon circumstances).

3.2.2 The Container Re-handle

The two objectives when planning container-ship stowage is to arrange cargo so that:

- the number of containers handled during unloading is kept to a minimum;
- the time spent in port is kept to a minimum.

This section attempts to illustrate this important objective. In order that the stowage problems associated with multiple-port container transportation can be illustrated, the following example (illustrated in Figure 3-1, Figure 3-2 and Figure 3-3) has been taken from the voyage made by the Sirius container-ship. The example shows the state of bay 31: inbound from Hamburg (illustrated in Figure 3-1); with the discharge containers removed (in Figure 3-2); the final outbound state after all containers have been processed at Antwerp. [12] The example is restricted to a single bay for simplicity. (Bay Plans, and their interpretation, are explained later in Section 3.3.2.3; a Bay Plan indicates where in a bay containers have been placed, and colours are used here to indicate the destinations of the containers.)

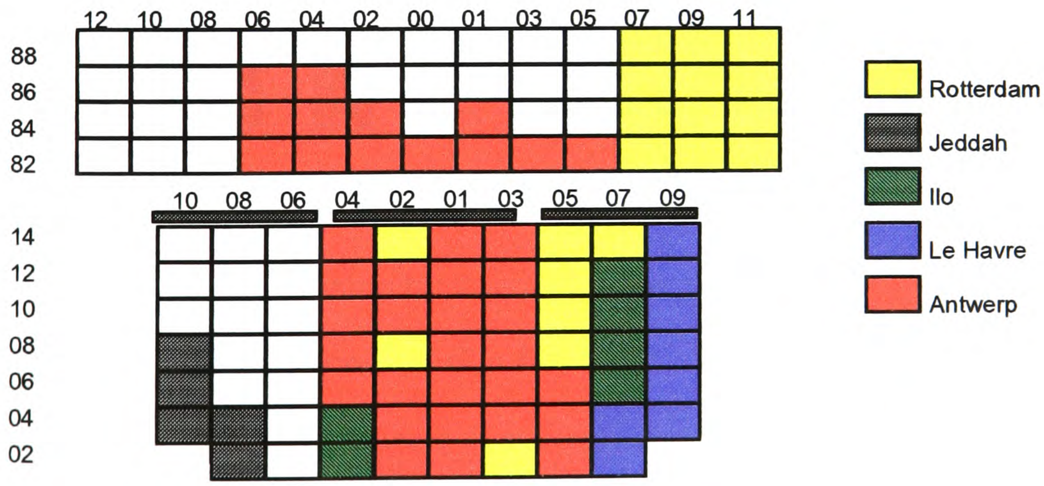


Figure 3-1 Bay 31, Inbound from Hamburg

In the example of a Bay Plan in Figure 3-1, a number of container locations, some filled, are shown for locations above and below deck (above which the numbers 10, 08, 06 *etc.* appear). It can be seen from Figure 3-2 that little consideration has been given to minimising re-handles at Hamburg since 18 containers have to be moved to allow access to containers due to be discharged at Antwerp. Since three containers destined to be discharged at Antwerp have been stored under the right hand hatch-lid, all those stowed above this cover must also be removed, or re-handled, to permit access. These removed containers are restowed later in the loading cycle.

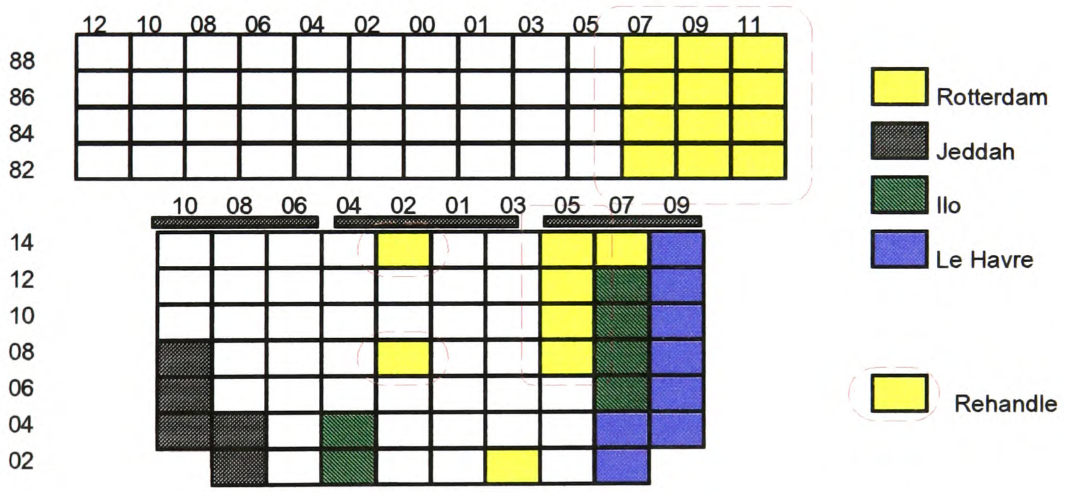


Figure 3-2 Bay 31 with discharged containers removed

All those containers that must be moved to allow the Antwerp containers to be discharged have been outlined in red in Figure 3-2. In this example 18 containers have been re-handled, in addition to a seemingly needless extra removal of a hatch cover. Had the containers been stowed differently, all of these re-handles could have been avoided.

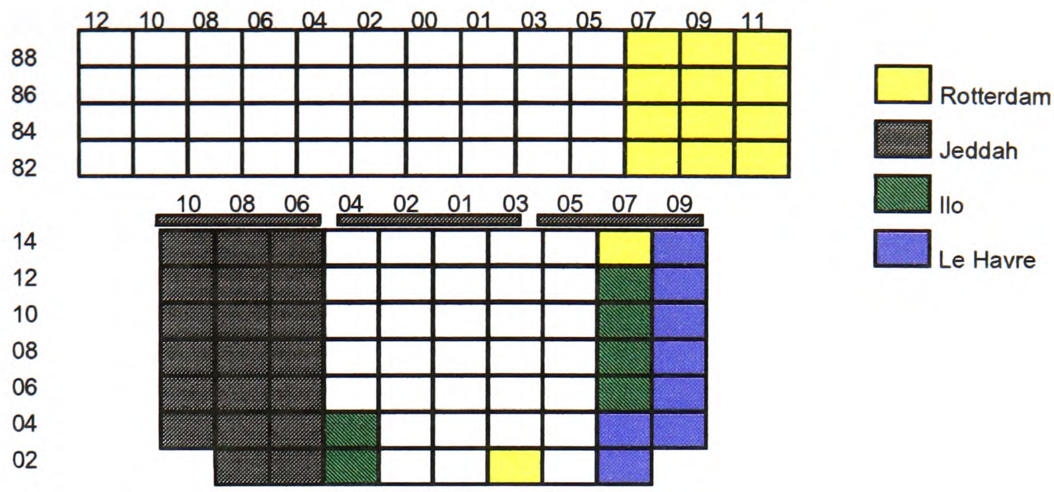


Figure 3-3 Bay 31 after completed loading process

This real example of an outbound Bay Plan (shown in Figure 3-3) clearly shows that the re-stows have been stowed elsewhere, perhaps to improve the overall stowage pattern of the vessel.

3.2.3 General Description of Stowage Planning

As was explained in Section 3.2.1, stowage planning can be thought of as being divided into two stages, termed by this author as strategic pre-planning and tactical planning. Strategic stowage pre-planning consists of a generalisation of the cargo unloading and loading sequence and allocations within the vessel in each of the ports on the route, so that:

- no ship stability and stress constraints are violated;
- utilisation of cargo-space is maximised;
- transport costs are minimised.

Planning has consequences for the ship operator (in terms of vessel space utilisation and efficiency) and the terminal (in terms of the costs of unloading and loading that are passed on as charges to the ship operator). Effective strategic pre-planning and tactical planning of the stowage of containers within container-ships reduces transportation costs and maximises vessel cargo-space utilisation, whereas effective planning of container-terminals reduces material handling, and by implication transport costs. These two very similar operations are often in conflict with each other since they represent the factors that influence the profitability of two, usually separate, bodies. Due to this, a planner is, by necessity, an extremely skilled individual.

The long-term decisions made by the strategic pre-planner affect the specific placement of cargo by the tactical planner (be it the same individual making specific container allocations or an individual at the container-terminal making the same decision). The same individual will often perform both the strategic pre-planning and tactical planning tasks and for this reason is referred to, simply, as the planner. The ship-operator and terminal planners must have a working knowledge of ship stability and container stowage constraints, and also must be able to conceptualise the best stowage pattern from the seemingly endless number of possibilities. The ship operator's planner must be familiar with the geometry and any special operational problems relating to both the vessel and the route in question.

In contrast to the ship-planner's required knowledge, the terminal-planner must be familiar with the terminal and any other considerations that influence the efficiency of the loading/discharge operation and, hence, the attractiveness of the port to ship-owners. Acquiring the knowledge required to be a truly effective planner can take many years. Planners knowledge is only partly explicit, the true expertise being almost wholly intuitive built up through experience.

The pre-planning process usually takes place at a centralised location, but can often take place at the container-terminal itself. Strategic pre-planning has to take into account the expected loads to be placed on the vessel at many subsequent ports in a ship's voyage. This planning ahead is made possible through the use of booking information, statistical forecasts and actual information about cargo at each given port. Booking information can be very limited, since although container numbers are provided, container weights are often omitted. Forecasts, and frequently best guesses, are included in planning and refer to cargo that may not eventually be loaded. Forecast information is often quite limited. Actual information refers to cargo that is sitting at the current berth, ready to be loaded.

The tactical planning of cargo stowage is left to the last moment since a vessel may often have to accommodate last minute changes to the (expected) stowage-plan. Such changes to the stowage-plan might be due to new cargo arriving, or expected cargo not arriving, at the container terminal. An exception to this is the requirement that hazardous cargo to be booked well in advance. This general readiness to accept cargo with little or no prior notice contrasts with the earlier days of shipping when cargo had to be at the pier well in advance of the vessel arriving. [12] This change has been made largely due to the highly competitive nature of the market and the resulting need to maximise cargo space utilisation.

Due to the above mentioned commercial pressures, the planner must deal with a best estimate of what the actual cargo will be. The first step taken by the planner is to determine which containers will have to be discharged from the container-ship. These containers fall into three categories:

- those arriving at their destination port;
- those that are moved to facilitate access to the former;
- and those moved simply to improve the overall stowage pattern of the vessel.

A container in either of the last two categories is known, interchangeably, as either a re-handle or a *re-stow*. Containers that block access to others that are to be unloaded first are known as *over-stows*. Although one of the main objectives when planning the stowage of cargo within a vessel is to minimise the number of unnecessary movements of containers at a port, very often there are occasions when this constraint is ignored. Elimination, or effective minimisation, of re-handles may not always be possible due to limitations upon where cargo can be stowed, perhaps because of hazardous cargo placement constraints, and the need to maximise utilisation of cargo space.

For a given port, the unloaded stowage configuration (the stowage configuration after discharge but prior to loading of containers at the port) is determined. Then the planner must prepare a target departure loading plan (the desired plan for the vessel on leaving the port). This loading plan must satisfy a number of constraints and stowage objectives. No one single rule is applied when trying to achieve the optimum stowage of a vessel. The fundamental objective when planning the stowage pattern of a vessel is to minimise vessel turnaround time. However, it is not sufficient simply to minimise the amount of time a ship stays at the current container-terminal, as decisions there will ultimately affect container-terminal turnaround time further along the route. The problem is therefore to find the loading arrangement of containers that minimises the total handling cost.[13] In other words, it is desired that the total number of container movements, across an entire voyage, be as small as possible. A sequence of actions performed by the planner when planning cargo stowage might be [9]:

- Prepare: About four days before arrival, the import section of the profile is prepared and summarised.
- Plan: About two days before arrival, the debarkation section of the profile is prepared and summarised. Specific instructions for where special cargo, such as hazardous, reefers, out-of-gauge and uncontainerised cargo along with generalised block-stow instructions for standard cargo (that may well be added to during the time the ship is being loaded) is given to the container-terminal.
- Check: About 12 hours after departure from the port of discharge, the planner checks the actual performance against the plan and updates accordingly.
- End of voyage: About 12 hours after departure from the last port of a rotation, a final summary report is prepared that includes statistics about performance and cargo carried.

The general guidelines for planning stowage are:

- protect the crew;
- protect the vessel;
- protect the cargo;
- limit time at port and maximise vessel utilisation. [1]

The first three guidelines are clearly related. Protecting the vessel must include considerations for stress and stability. Protecting the cargo includes not placing hazardous chemicals in close proximity to sensitive goods. The last two guidelines largely affect cost effectiveness. As well as these guidelines, there will also be a variety of physical constraints that restrict the placement of containers, due to restrictions on the size of container that can be placed in a specific location on a ship.

The vessel must be safe to sail at all times and in order that this be accomplished a number of factors relating to the integrity and stability of the vessel are adhered to. The cargo, stores and ballast tanks of the ship directly affect factors relevant to the safety requirements for a ship's condition both at sea and in port. These factors include ship torsion, centre of gravity, bending, trim and heeling. There are guidelines concerning these statistics that are different for when a ship is in port and when it is at sea. [7]

3.3 Stowage Considerations

This section illustrates and contrasts the different perspectives, and priorities, of container transportation for the shipping operator and the container-terminal. An in depth discussion of what documents are used to facilitate the stowage planning operation is offered indicating how each helps the planner's task.

3.3.1 Introduction

Both the shipping line and the container-terminal have similar, but often conflicting, stowage considerations (introduced in Section 3.2.3). The container-terminal planner's view of the tactical cargo stowage planning problem is, given a set of containers in the container-yard and a set of locations on board ship, to determine the allocation of containers to locations and the corresponding loading sequence so that all constraints are satisfied and material handling costs are minimised. These constraints include:

- ship stability;
- requirements for the storage of hazardous cargo;

and such special storage requirements as:

- refrigerated units;
- deck strength limits;
- container stack height limits;
- and container length restrictions.

The degree to which the ship-operator is concerned about individual container placements upon the container-carrier varies greatly. Some ship-operator planners present very precise stowage plans to the container terminal and others provide instructions that give a variable degree of choice. The ship-operator may want specific container placements to be made that prevent optimal container load, thus being in conflict with the container-terminal objective of minimising material handling costs.

3.3.2 Documentation for stowage planning

The following section introduces the different documents used to plan container-stowage.

3.3.2.1 The General Arrangement Plan

The *General Arrangement* (an example of which is shown in Figure 3-4) is a simplified, small-scale, vertical longitudinal section through the centre of the vessel, viewed from the starboard side. It shows the positions of the:

- Hatches;
- *tanks* (ballast tanks, fuel tanks *etc.*);
- *non-cargo spaces* (areas of the ship where cargo can not be stowed);
- *accommodation block* (an area of a ship where the crew live);
- *engine room* and so on.

It also shows where and how containers are stowed, in fore and aft orientation, above and below the *weather deck* (a term used to describe the surface area of a ship that is exposed to the weather), as a series of stacks. The general arrangement is also known as the *General Plan* or *General Stowage Plan*. [3]

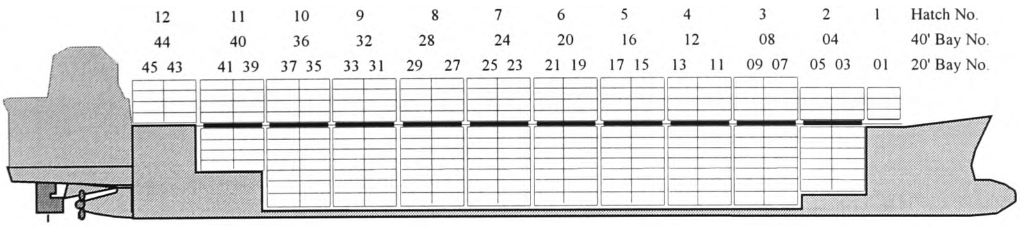


Figure 3-4 General Arrangement

The General Arrangement provides information that can help when planning the ship operation, specifically:

- The location of each hatch; this shows whether cranes can work adjacent bays at the same time, thereby speeding up loading and discharging; cranes will not be able to operate simultaneously on bays located side-by-side.
- The position of the accommodation block and engine room, which is important when considering crane positioning and hazardous container stowage.
- The spaces between bays above deck that permit personnel access for the manual operation of lashing.
- Which bays are restricted to one size of container only; the General Arrangement shows which bays have restrictions about what length container they can accommodate.
- Stowage locations that are only suitable for *empties* (a term commonly used to describe empty containers); these are indicated by dotted outlines on the General Arrangement. Locations in which loaded containers can not be placed.

- The maximum number of containers that can be stowed *athwartships* (transversely across the ship, from one side to the other) above-deck is sometimes shown on the General Arrangement, printed in a triangle above the on-deck stowage positions.

It can be seen that the General Arrangement gives a great deal of information about the vessel and its container stowage capability. However, few General Arrangements show how many containers can be stowed across the vessel at each level above and below deck. The only piece of information it provides concerning the number of containers that can be stowed is the maximum number at the widest level. The number stowed athwartships clearly depends on the shape of the hull and the presence of restricting features such as tanks, and so will not always be the maximum at the *widest* level.

3.3.2.2 The Outline Plan

In an O*utline Plan* (illustrated in Figure 3-5), the container stowage stacks of the entire ship are shown in more detail, in the form of a series of vertical transverse sections. Each section, or bay, is viewed from aft. Usually, a small version of the general arrangement is included in one corner of the outline plan. The Outline Plan is also commonly known as a *Single Letter Plan* or just *Letter Plan*. This second type of plan displays the information missing from the General Arrangement, namely the specific number and locations of all stowage slots. The information is displayed in the form of a series of cross-sectional views of the bays viewed from aft. Each stowage location is shown as a small box. The shape of the stowage in each bay, dictated by the ship's hull and the presence of tanks is clearly indicated.

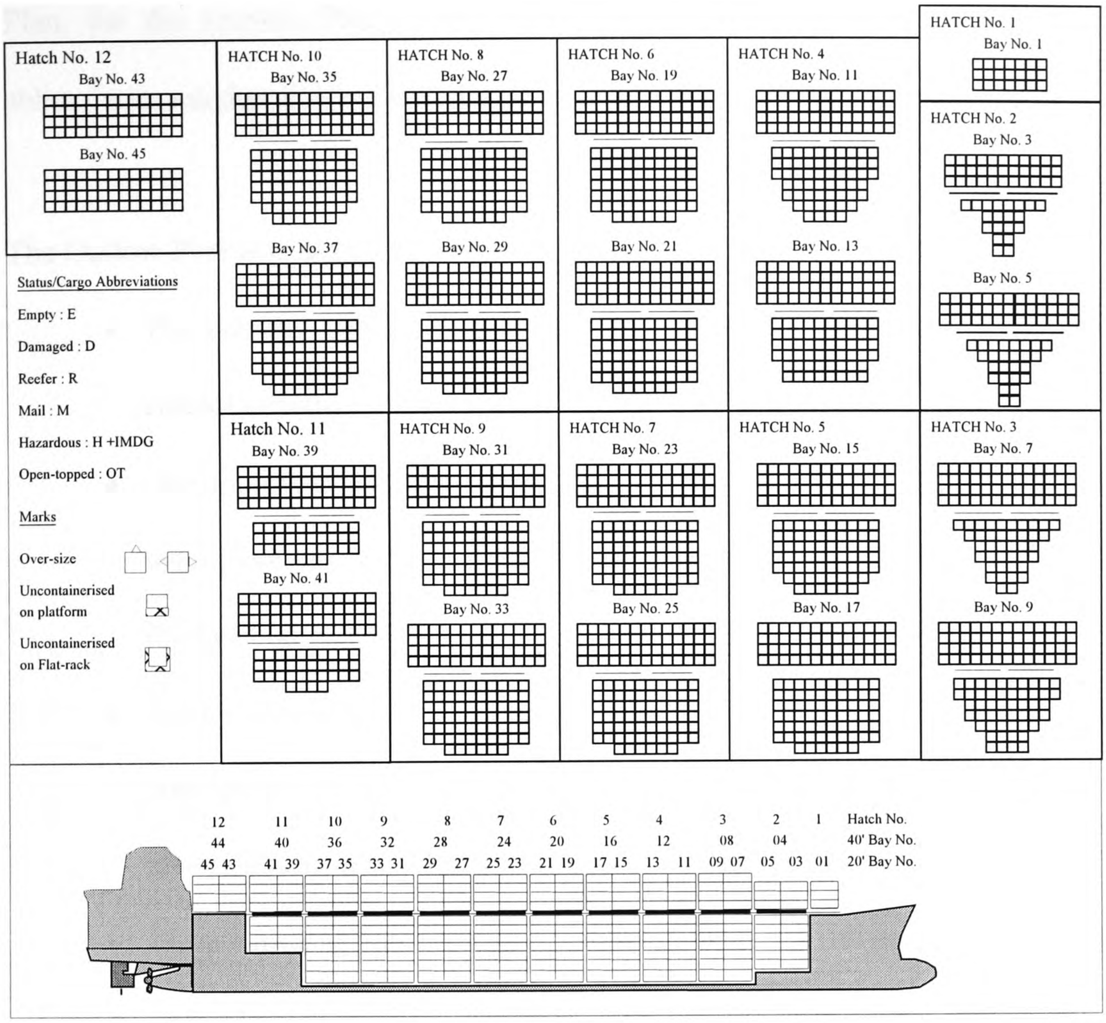


Figure 3-5 Outline Plan

The Outline Plan document is central to the operations of planning ship loading and discharging. The container positions can be shown in a variety ways:

- A single letter (or sometimes two letters) may be inserted in each occupied slot to indicate the port of discharge (*e.g.* 'P' for Port Kelang, 'K' for Hong Kong, 'S' for Singapore *etc.*).
- The square is coloured according to an agreed colour code (*e.g.* light green for Port Kelang, purple for Hong Kong, red for Singapore).
- A combination of the above.

(The use of letters gives rise to the alternative names, Single Letter Plan or Letter Plan, for the Outline Plan.) Whichever method is used, a code table of port abbreviations and/or colours is included somewhere on the outline plan.

The Outline Plan also indicates a range of other useful planning features:

- The relevant container slots can be marked with symbols to show where special containers (explained in Section 2.3) will be or are stowed;
- Over-height containers can be indicated by drawing a small triangle over the square (as indicated in the status/cargo abbreviations in the Outline Plan of Figure 3.5);
- Over-width containers can be indicated by drawing a small triangle on the appropriate sides of the square (as indicated in the status/cargo abbreviations in the Outline Plan of Figure 3.5);
- The position of power supplies can be indicated by an appropriate symbol;
- Refrigerated containers can be indicated;
- The presence and type of hazardous cargo can be indicated using appropriate codes;
- Uncontainerised cargo carried on flat racks can be indicated;
- Other special container characteristics can also be indicated;
- The positions of all hatch-covers are shown as thick lines between the above-deck and below-deck container slot squares on the bay profiles;
- The planner can see at a glance how many hatch-covers will have to be removed before under-deck containers can be moved;

- The planner can see how many above-deck containers need to be removed before a hatch cover can be accessed;
- The Outline Plan shows exactly how many containers can be stowed in each bay.

The actual coding system used to indicate the above features vary from operator to operator, but will be shown somewhere within the Outline Plan. The Outline Plan makes it clear that the containers are stacked in vertical rows along and across the ship. In the very largest cellular vessels (the Post-Panamax generation of ships) there can be as many as 16 rows athwartships above deck and 14 below deck. [3] In the more common third-generation or Panamax vessels there are typically 12-13 rows carried athwartships on deck on most hatches but no more than nine across below deck. [*Ibid.*] The majority of cellular ships are smaller than Panamax and have fewer rows both above and below deck.

3.3.2.3 The Bay Plan

A bay plan is a detailed view of just one of the stowage bays from the Outline Plan. Sometimes, the above-deck and below-deck stowage positions are shown on one sheet, but often, separate sheets are used for the above-deck and below-deck parts of a bay. A complete Bay Plan for a ship will be a large document composed of many sheets, each of which will be similar to the generic example shown in Figure 3-6.

Voyage number:______ Date:______ Port:_____/______
Discharging/Loading

210814	210614	210414	210214	210014	210114	210314	210514	210714
210812	210612	210412	210212	210012	210112	210312	210512	210712
210810	210610	210410	210210	210010	210110	210310	210510	210710
210808	210608	210408	210208	210008	210108	210308	210508	210708
210804	210606	210406	210206	210006	210106	210306	210506	210706
	210604	210404	210204	210004	210104	210304	210504	
		210402	210202	210002	210102	210302		

Bay No. 21
Under deck
8' 6"

Figure 3-6 A Bay Plan

While the General Arrangement and Outline Plan are both useful when planning, by providing detailed descriptions of the layout and capacity of the vessel, they do not provide enough space in each of the stowage slots for inserting all the necessary details of the container located in it or to be loaded into it. These two documents are often used to indicate the broad allocation of groups of slots to particular ports of discharge, and the location of containers of dangerous goods and other special containers. A larger and more detailed plan of each bay must be provided for the planning and supervising of the actual stow for a loading operation and the detailed sequence for discharge. The Bay Plan gives an expanded view of each bay shown on the Outline Plan, and is provided on a separate sheet. There will be three sheets for each bay that can accommodate either 20' or 40' containers.

When planning is complete an adhesive label is made for each of the containers, and these labels are attached to the stowage slots on the appropriate Bay Plan. Each Bay Plan is large enough to provide sufficient space for a considerable amount of information about each container.

The information contained on the label usually includes the following (and a typical example is shown in Figure 3-7):

- the slot address;
- the container identification code;
- the port of discharge;
- the port of loading;
- the gross weight of the container and cargo;
- the container type;
- the cargo contained within if of a special type;
- the dimensions of the container;
- and the operator's code, this may be different from the owner's code.

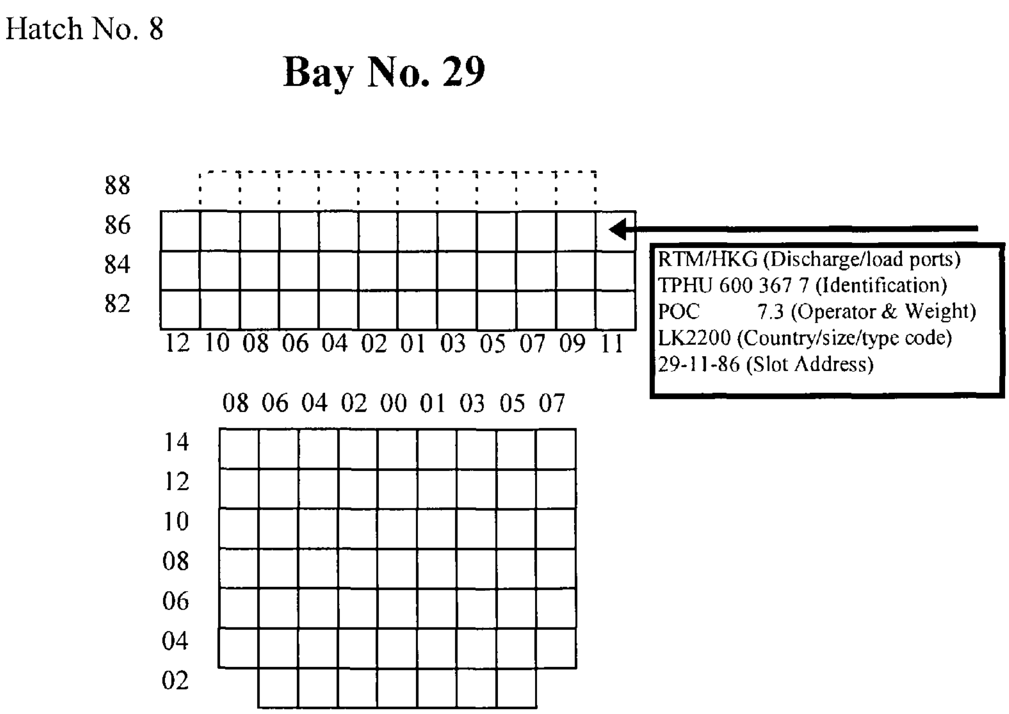


Figure 3-7 Slot label example

To illustrate the usefulness/use of this information, consider the following situation. A Bay Plan, such as the one in Figure 3-7 [3], may be received at Rotterdam and then used there to plan and control the container discharge operation. The label shown then indicates that the container in that slot was loaded at Hong Kong (HKG) and is to be discharged at Rotterdam (RTM). Earlier in the voyage of the ship, that same label would have shown the loading staff at Hong Kong which slot to place the container and its ultimate destination. The container's identification code shows that the container belongs to Tiphook Containers (TPHU) but that it is under lease to P&O Containers (POC). The container was registered in Sri Lanka (LK), it is 20' long and 8'6" high, and of a general purpose, basic type (2200) and weighs 7.3 tonnes. It is located in Bay 29, Stack 11 and Tier 86. Finding a container within a cellular container-ship, given such information, is a straightforward task (as explained in Section 2.3).

Non-containerised cargo can also be indicated on the bay plan, along with a variety of other *specials* (a term used to describe cargo that requires special handling, usually distinct from hazardous types), some of which follow:

- crates of machinery not contained in a container;
- a number of folded-down flat-rack containers;
- open topped;
- high-cube (a container 9'6" high); the growing use of such containers is discussed in Appendix C;
- top stow;
- and under-deck.

The bay plan also contains information designed to assist the Planner:

- make required intact stability calculations;
- ensure that stack maximum weights are not exceeded;
- ensure that stack maximum heights are not exceeded.

The total weights carried in each tier and stack, as well as the total for the bay, will be shown on the Bay Plan. Factors are shown on the Bay Plan, at the top of stacks, in order that the Planner can calculate vertical moments and other data by applying them to the stack weights.

3.4 Stowage Planning Guidelines

The following points outline the main constraints and guidelines, common to most operators, that must be considered by planners during the stowage planning process for an individual port. [12] The list is not comprehensive, but is sufficient to illustrate the large variety of factors that require consideration.

(i) The number of times a container must be re-handled before discharge is to be minimised. The exact cost of a re-handle varies from terminal to terminal and from operator to operator. Large savings can be made by reducing the number of re-handles although constraints (described in Sections 2.3.5, 2.3.6 & 3.3) will usually make it impossible to achieve zero restows.

(ii) Ballast can be used to correct stability problems, minimise torsion and shear forces and bending moment stress and help achieve a desired trim (introduced in Sections 2.3.5 & 2.3.6). However, ballast should be minimised since the vessel is in effect carrying dead weight, that directly affects the efficiency of the vessel.

(iii) Structural stress is constrained by guidelines set down by the Classification Society. Placement of containers along the ship affects weight distribution and, as a consequence, causes stress. The buoyancy distribution has its peak amidships. A rule of thumb for reducing the value of the bending moment is to fill the positions amidships. However, if that section is filled with the heaviest containers, the deck may buckle. Conversely, if the heavier containers are placed exclusively at the ends, the vessel bottom may buckle. To minimise torsion stresses, cargo must be stowed evenly across the vessel. When the vessel is discharged, cargo is unloaded in such a way that no torsion limits are exceeded. The vessel must meet minimum static-stability requirements. When the cargo is not homogenous, then the centres of gravity of the contents of all containers must be used when performing intact stability and stress calculations. The whole process of tactical stowage planning can take a considerable amount of time, dependant on the size of container-ship, when performed by hand.

(iv) Due to safety and efficiency constraints the vessel must operate as close to zero trim as possible. If zero trim is unattainable, stern trim is preferred to bow trim so that propeller immersion is maintained and slamming force is reduced (described in Section 2.3.5.4). [7] If a ship does not have an acceptable trim and stability, it can not leave port. Vessels are equipped with ballast tanks that allow some adjustment of heel (see Section 2.3.5.2) and trim during and after the cargo operations are performed.

(v) Crane, manpower and cargo-space utilisation is to be optimised. The stowage-planning problem is separate from, but closely related to, the problem of planning the stowage sequence. The former is concerned only with the final stowage plan whereas the latter determines the order in which containers are loaded and unloaded. The two are related since the stowage plan determines the number of over-stows and, hence, the number of additional crane movements required when performing cargo stowage operations.

(vi) A stowage plan that minimises over-stows may itself be inefficient if the number of moves made by a crane and the distance travelled by it is excessive. Whereas it is sensible to group together cargo with the same destination in the same bay, a good disposition of this cargo between bays will multiple cranes to work simultaneously (illustrated in Figure 3-8). An optimum separation of four bays between cranes is required to facilitate simultaneous operation.

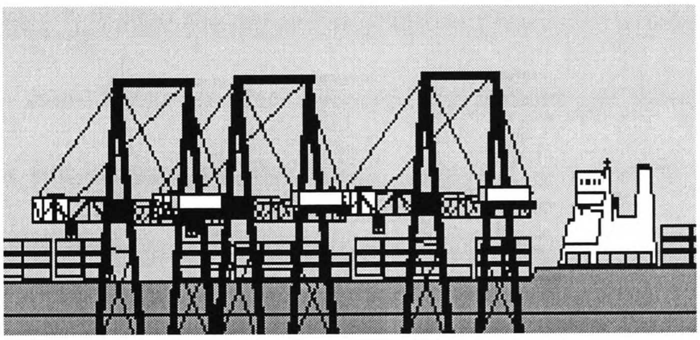


Figure 3-8 Container-ship with three cranes in operation

This parallelisation of the loading and unloading process will permit a faster turnover of container movements to take place. Some vessels have their own cranes, either of the familiar crane type (consisting of a pivoted boom rotating about a vertical axis with lifting gear suspended from the end of the boom) or gantry type (a bridge like framework used to support a travelling lifting gear suspended from the structure). The interaction of the cranes requires consideration when planning a stowage pattern.

(vii) Vessels normally have 40' units placed on top of 20' units. Where 20' units are of a different height, 6" filler pieces can be used to bring the containers up to the required height. Stacks may not be completely filled due to stack weight limits, so stowage planning should ensure the maximum use of TEU and hence minimise the amount of lost cargo spaces. Ideally, only one discharge port's cargo should be stored under a single hatch (*e.g.* Hamburg). If this is not possible then the space should be taken up by cargo for another port with the furthest distance to travel (*e.g.* Hong Kong). For example, this would then allow room for additional cargo for Hong Kong to be stowed in the space left after the cargo for Hamburg has been discharged.

(viii) Cargo should only be placed in appropriate areas of the ship, although this is not always possible. For example, some cargo can only be placed in areas specifically allocated for its use, (*e.g.* hides may only be placed in areas of the vessel that have been specially treated). Each of the two types of *Reefer* unit (refrigerated container either independently powered or by the ship via a dedicated power outlet, see section 2.1) available should be stowed according to the appropriate rules and stowage requirements. On vessels that support this type of container, care must be taken to segregate the reefer commodities, so that tainting does not occur. Reefer placement follows some general rules. Priority is given to placing reefers in designated reefer slots. Where possible, 40' reefer containers should be placed in stowage slots where only one reefer slot is used, rather than occupying two 20' reefer slots. Reefers should be stored away from locations that give off radiant heat, such as the Engine Room and Fuel Tanks. Empty reefers should occupy standard locations, *i.e.* not locations designated for reefer storage.

(ix) Containers with hazardous cargo invariably must be separated by a minimum distance from other containers also containing hazardous cargo. Violations of the code governing the placement of hazardous cargo carry severe penalties. Stowage is planned so that hazardous cargo is separated according to the segregation rules. Where conflict with the segregation table does not occur, hazardous cargo should be stored on deck. Hazardous cargo should be stored away from crew accommodation.

(x) Wet hides and wet salted hides tend to leak and give off a pungent odour. The residue is a brine solution that is highly corrosive, highly pungent and which fouls *bilge* (the parts of the vessel's hull where the sides curve inwards to form the bottom between the lowermost floor and the hull) *systems* used to pump out *bilge water* (the dirty water that collects in a vessel's bilge). [12] Therefore, hides can only be stowed within cells that have been specially treated to receive them. Additionally, hides must always be at least two cells horizontally away from reefers or open topped containers, and three bays away from crew accommodation. Hides are not allowed above or next to foodstuffs.

(xi) Flat Racks are one of the most common forms of transporting fresh foodstuffs. Certain foodstuffs that give off gas, such as onions, must not be stowed next to reefer units with chilled cargo. Ventilation should be provided for this type of cargo.

(xii) Flat racks are stored in nests of six in one 20' cell. All flat racks should, where possible, be stowed under deck regardless of length, height and type. If there are less than six flat racks within one slot, problems will occur when stowing containers on top.

(xiii) Out of gauge containers should be placed at the top of stacks as this will minimise interference with adjacent slots. Similarly, over-height containers should also be placed on top of stacks.

(xiv) *Fantainers* (a name given to containers that are ventilated by an internal fan, see Section 2.1) must be stowed near to reefer outlets in order that use can be made of the power points associated with these slots.

(xv) Empty and open top containers should usually be placed on top of stacks.

(xvi) Gradation in weight should be observed - that is, heavier containers should generally be placed at the bottom, and maximum allowable stack weights should not be exceeded. 20' containers should only go into 20' designated cells where possible. 40' containers should only be placed into 40' designated cells where possible. Cells are built normally for standard 20' containers, so if a 40' container is to be stowed and there are no more free 40'-only cells, then two corresponding cells will be occupied where the structure of the vessel permits.

(xvii) When allocating cargo between more than one bay, loads for a particular discharge port should be split evenly between bays in order to minimise potential stability problems that would occur when the cargo are removed and improve crane deployment

(xviii) Sometimes, so that vessel utilisation is maximised, containers may be stored in areas that are difficult to access at certain destinations. For example, the berth at which the ship docks may not have cranes that can access an extreme part of a vessel. Access to containers must be weighed against vessel utilisation.

(xix) The effect that loading hazardous cargo has upon TEU utilisation should be minimised. Placement of hazardous or special cargo may make some slots unacceptable stowage locations for other cargo types. Therefore, some cargo may have to be left behind if hazardous containers have been placed without due care and attention.

(xx) Poor block stowage of cargo intended for the same destination results in having to access an excessive number of hatches during unloading. Therefore, stowage should be planned so that hatch usage is efficient.

(xxi) Access to some containers (such as hazardous types) may be required during a voyage and these should be stowed accordingly. (In most cases this means on deck.)

(xxii) Stack height restrictions are to be observed and special consideration is to be given where crane height may be less than normal stack height.

(xxiii) The cargo weight distribution should be within acceptable bounds set by metacentric height (GM) requirements, dead-weight limits, draft restrictions, and hull strength limitations.

As a result of this diversity of factors influencing the stowage planning of containers it is not an inconsequential problem to determine a pattern of stowage that is close to optimal whilst meeting all these stowage constraints. In addition to considering all the above stowage constraints, the pre-planner must focus upon arranging the containers for optimal port efficiency and vessel utilisation.

3.5 Summary of stowage planning

As little time as possible should be spent in port loading and discharging containers. This can be achieved by reducing the distance travelled by cranes, the number of container re-handles and amount of hatch-cover lifting. Increasing cargo-ship utilisation means maximising the number of containers carried, minimising ballast and optimising trim. Maximising the number of containers carried increases turnover making the business more profitable. Minimising ballast and optimising trim reduces running costs. Since many of the above factors may be mutually exclusive, trade-offs must be decided upon. Such decisions require a complete analysis of any decision weighing up the implications and associated costs.

Since deep-sea container-ships serve many different ports on each voyage (Figure 3-9 shows the route taken by the Sirius - a 2500 TEU container-carrier [63]). The difficulty in achieving good stowage increases the longer the ship is at sea. In a multiple port trade each port may load containers for several different destinations. The space available on board the vessel for these containers is equal to the space made vacant by discharging inbound containers and those locations that were vacant to begin with. This decrease in stowage flexibility often leads to a fragmentation of the cargo space with containers with the same destination being spread in an increasingly haphazard manner around the ship.

The progressive degradation of stowage efficiency is directly attributable to the vessel's activity in prior ports. The resulting degradation of on-board cargo arrangement, persists until the vessel is completely discharged, if ever. There are two alternatives to avoid this stowage degradation. The first, undesirable solution, is to completely discharge and reload the vessel at each port. The more practical but much more difficult alternative is to stow the containers at each port in a fashion that will minimise problems at future ports. In order to do so, all containers to be loaded at future ports must be considered in the development of the stowage for each individual port. In this way, a correct formulation of the strategic pre-planning problem involves all ports and trade simultaneously.

3.6 Problem Scope

In order that the scope of the problem described in the previous sections can be understood, this section describes a typical voyage made by the Sirius, a 2500 TEU cellular container-ship. [63]

3.6.1 Sample Voyage

The following diagram (Figure 3-9) shows the route, taken by the Sirius cellular container-ship, from Europe to Japan.

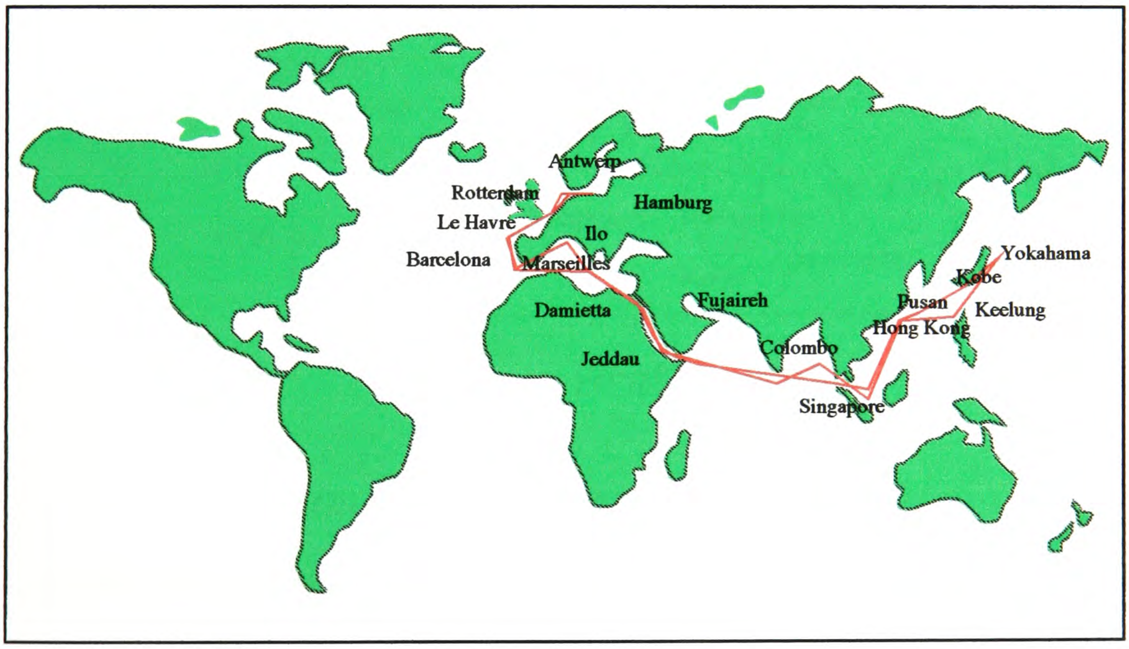
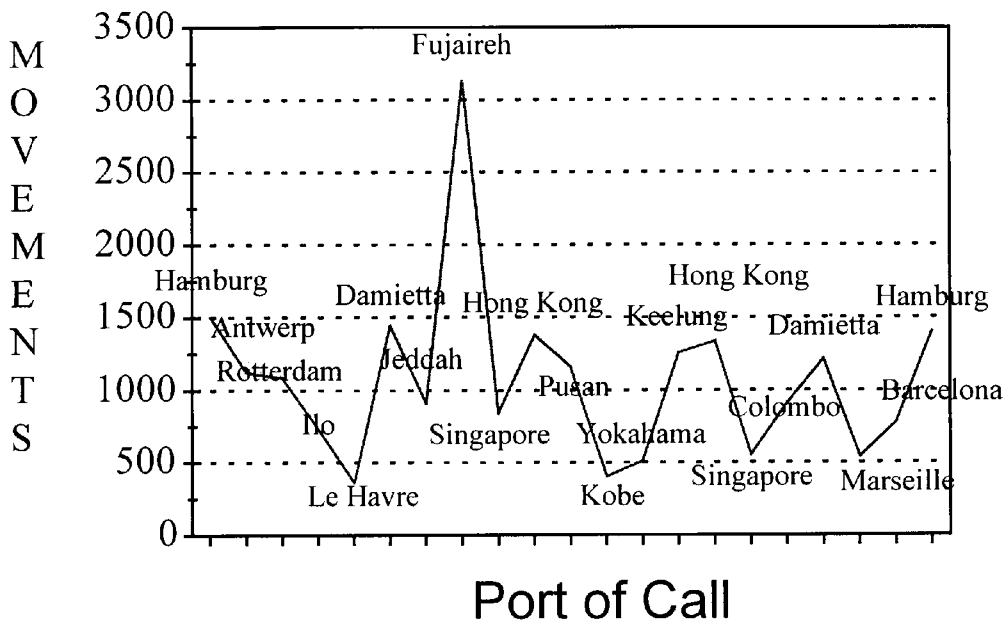


Figure 3-9 Example Port Rotation

3.6.2 Voyage Analysis

The total number of movements (see Graph 3-1) at each port includes all containers discharged, loaded and re-handled. (A re-handle is a movement of a container that is neither a load or a discharge and may occur either due to the need to access another container or in an effort to arrange the stowage of the ship more effectively; see Section 3.2.2.)

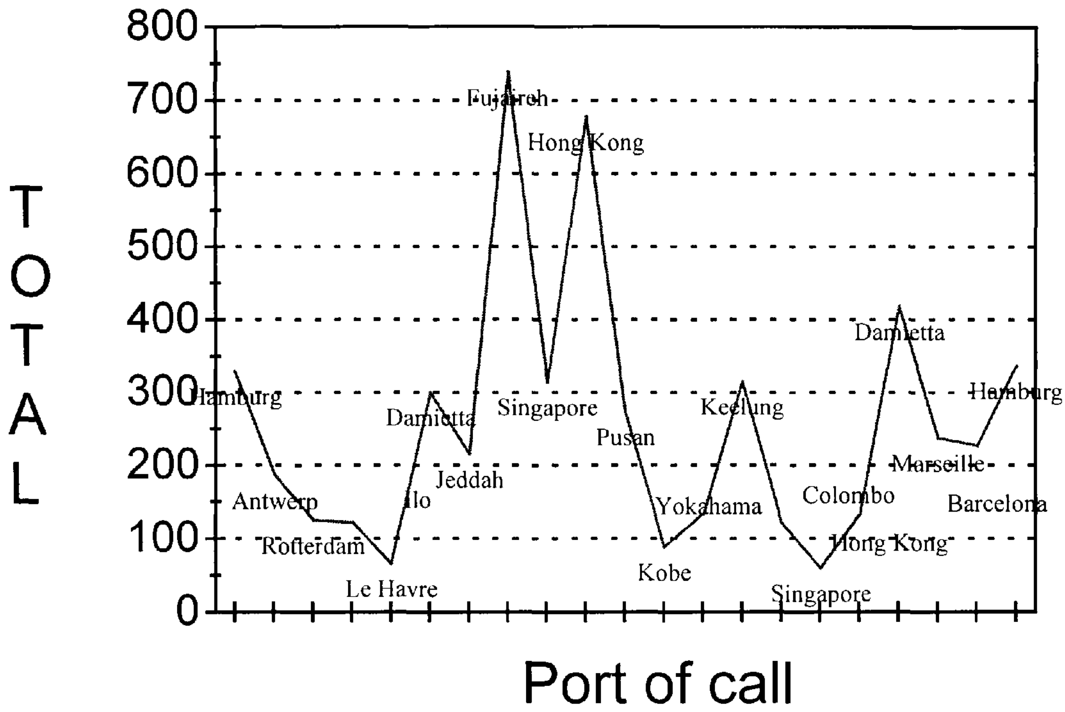


Graph 3-1 Number of Container Movements

It can be seen that the number of movements at each port varies considerably.

The stowage pattern of the containers loaded on the Sirius at each port will affect the number of movements at later ports. Container movements are normally kept to a minimum by the container-terminal, with the port having a large effect upon the precise stowage pattern of the vessel when it leaves port. The average number of containers discharged at each port is two hundred and fifty- eight. This breaks down as one hundred and eighty-one 20' containers and seventy-seven 40' containers.

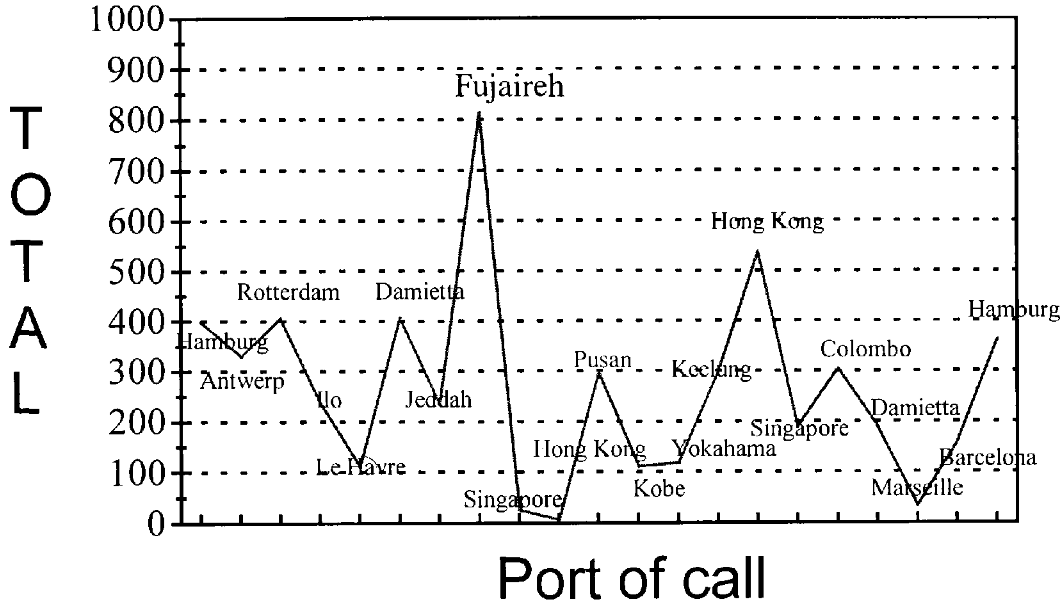
Although no precise figures that indicate how many non-standard containers were carried by the Sirius are unavailable, a review of all containers shipped indicates that two percent are 10', 24', 35', 45', 48' or 53' in length [14].



Graph 3-2 Number of Discharged Containers

The statistical break-down of lengths of containers carried world-wide indicates that sixty-two percent are 20' and thirty-six percent are 40' (the remaining two percent being of the non-standard variety indicated earlier). [14] An analysis of containers carried by the Sirius shows that seventy percent of containers are of the 20' variety and thirty percent are of the 40' class. This indicates that the Sirius data are representative of the world-wide transport of containers.

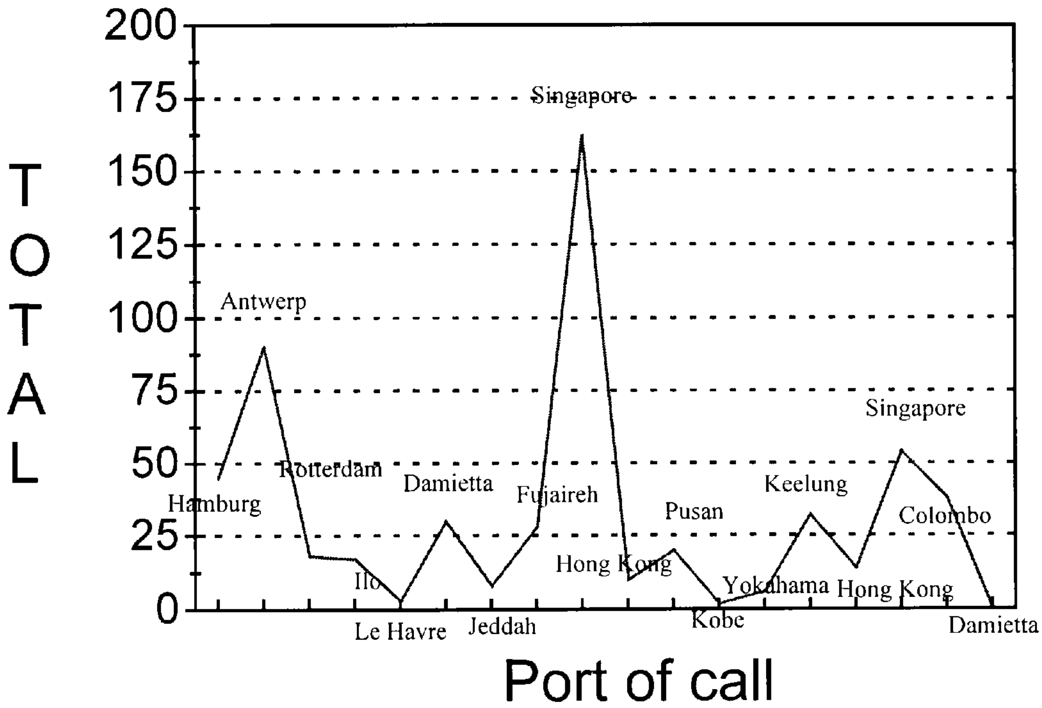
Of all the containers carried by the world container fleet 91% are 8' 6" in height (96% of 20' containers and 85% of 40' containers), the rest being less than 8' or great than 9' 6" in height. [14] For a ship such as the Sirius, perhaps as many as forty-five containers will be of a non-standard height or length (the standard size being 8' 6" in height and 20' or 40' in length).



Graph 3-3 Number of Loaded Containers

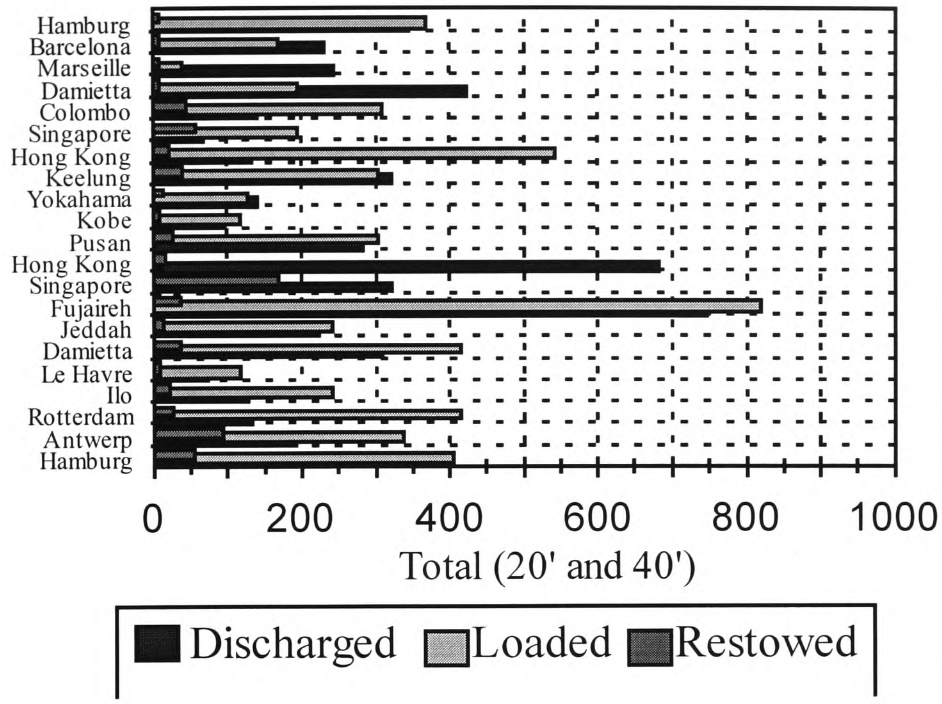
In an attempt to alleviate the problems caused by these non-standard sized containers, planners will generally place them at the top of stacks. In addition to the problem of stowing containers of a non-standard size is the problem of segregating hazardous cargo types [5] and providing electricity for containers (e.g. Reefers).

The number of restows or re-handles that take place at a port is usually a consequence of stowage decisions made at an earlier port of call. It is no coincidence that the number of re-handles increases dramatically at Singapore shortly following a visit at Jeddah where container handling charges are expensive (see Graph 3-4). The need to maximise vessel utilisation must be weighed against the need to minimise number of re-handles and port costs before deciding whether to take on board a particular container.



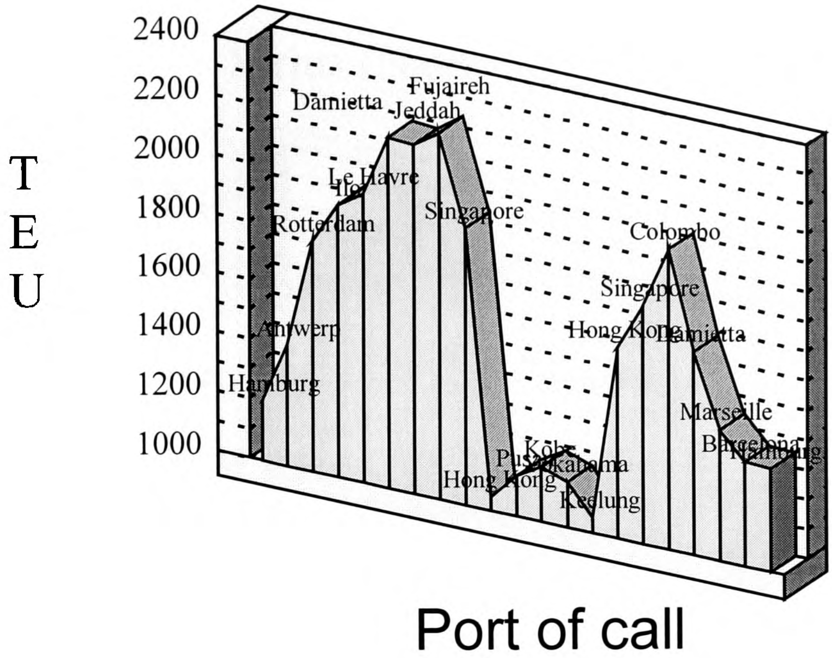
Graph 3-4 Total Restows

Due to the cost of re-handles varying at different ports, it may often be the case that a planner moves around the cargo of a vessel where there is no immediate need, in order that savings can be made further along the journey. A comparative review of total movements of containers is given in Graph 3-5. This indicates the scope of the cognitive process being exercised by the planner (where the number of restows is usually seen as a measure of the effectiveness of a given planner).



Graph 3-5 All Container Movements

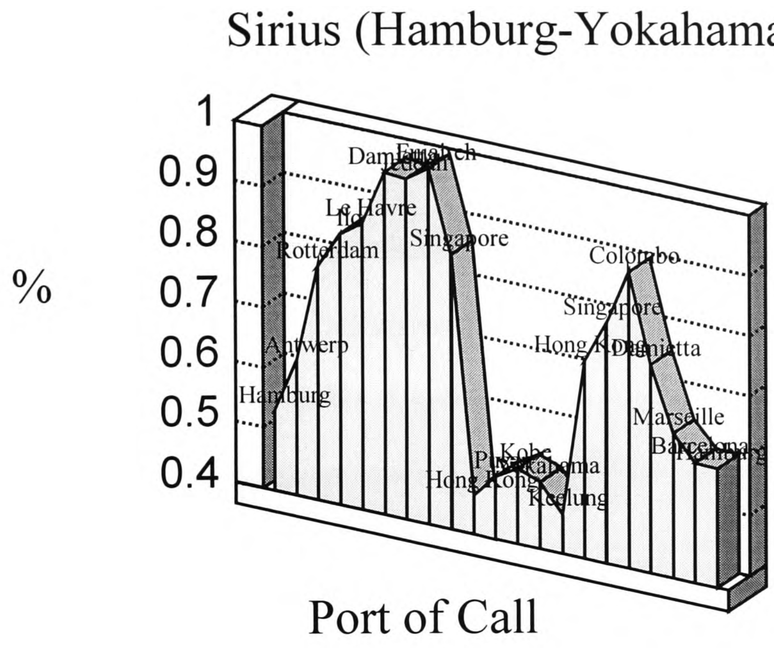
The following graph (Graph 3-6) gives the total number of 20' equivalent units (TEU's) carried by the Sirius at each of the ports visited.



Graph 3-6 Total Twenty Foot Equivalent Units (TEU)

The scale of the problem should now be clear. During the voyage a total of five thousand four hundred and twenty one containers were discharged, five thousand five hundred and sixty-three loaded with five hundred and seventy-seven containers being re-handled, giving a total of twenty-two thousand five hundred and forty-five movements of containers during the voyage.

Vessel utilisation is an essential aspect of container transport (as shown in Graph 3.7, which shows the same information on a different scale, to indicate percentages of vessel utilisation). The Sirius filled an average of seventy-three and a half percent of its total TEU capacity during this voyage.



Graph 3-7 Vessel Utilisation

An almost innumerable number of possible solutions are available to the planner. Each port will generate a number of feasible solutions, the implications of each will have to be explored at each successive port. The planner's problem can be split into two parts: in what order should containers be examined for loading, and where should each container actually be loaded. This is to say that the true combinatorial scale of the loading problem becomes clear when the order that containers are loaded and the actual placement of each container are considered. It is not feasible to attempt to exhaustively search every possible load sequence (as demonstrated by the discussion of Botter's [36] work described in Section 5.4) due to the problem being factorial in nature (having as few as 10 containers to load gives 3628800 possible load sequences [ibid.]). Since a typical voyage can contain up to sixteen ports, or sixteen different load sequences to determine, it is clear that exploring every possible solution in an exhaustive manner is impractical. Given a container-ship with a capacity of 2500 TEU with an available capacity of 500 TEU and a load list containing only 20' containers the number of possible container placements would be 500 when dealing with the first container on the list, 499 when stowing the second, 498 with the third and so on. After allowing for constraints upon stowage choices reducing the number of feasible stowage patterns the planner still has an immense combinatorial optimisation problem to solve.

Since it is clear that attempting to find an optimum solution by searching each possible state is impossible, some knowledge about the way the planner produces stowage plans must be introduced into the solution strategy. An algorithm that closely models the thinking process of the human planner must be used.

Standard search techniques can be used to solve the container carrier stowage problem. The loading of tankers using search is being explored [46, 47], but little exploration has been carried out using this option with container-ships. What research that has been carried out has included too many simplification processes, due to the combinatorial and computational size of the problem, with the result that the systems developed are of little use in the 'real-world'. The many factors of the problem have now been clearly introduced and the non-trivial nature of the problem demonstrated. The following chapter introduces general problem solving techniques and the issues associated with them.

4 ARTIFICIAL INTELLIGENCE

The research project detailed in this thesis centres around the use of techniques in *Artificial Intelligence* (AI). This chapter outlines the fundamental principles of AI, and is specifically intended for readers who are unfamiliar with the application of these techniques. In particular, the techniques of *Search* and the underlying theory behind so-called *Expert Systems* will be introduced; an understanding of these will be required when reading the critique of other authors' work in the area of ship-loading (given in Chapter 5) and the descriptions of the reasoning behind the work of this project (presented in Chapter 6).

4.1 Search and Artificial Intelligence

Search is a generic term encapsulating a variety of methods used to solve intelligent tasks.[15] Search plays a part in all aspects of AI [16,17,18,19,20] from *Natural Language Processing* [21,22,23] to *Machine Learning* [24,25,26,27, 28, 29, 30, 31]. Furthermore, and most importantly for this research project, it is the fundamental principle that permits the solving of complex problems.

4.1.1 Search and State-space

State-space is the formal description of a problem in terms of all possible alternatives. For example, in chess the state-space representation is the set of all possible positions and the rules for determining the moves. Knowledge about the domain is required in order that a good move and thus a good position can be determined and selected from the set of available possibilities.

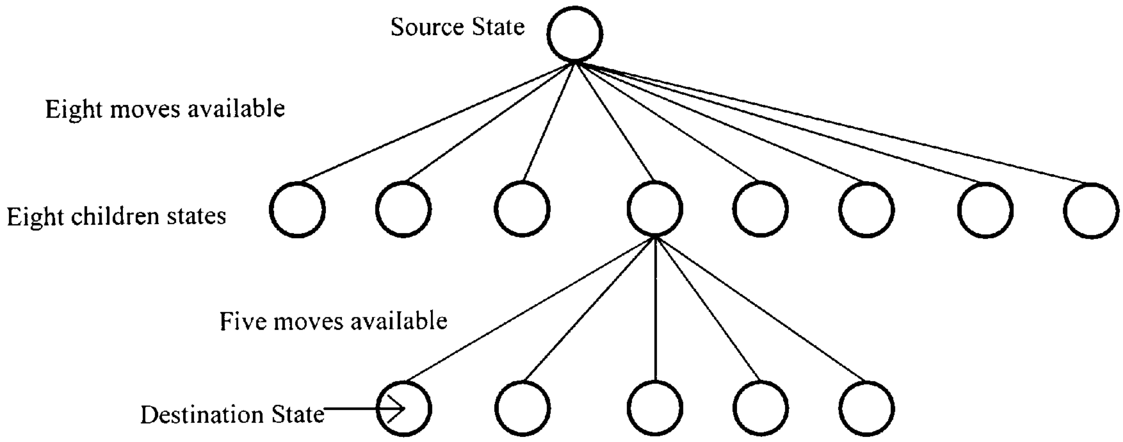


Figure 4-1 State-space represented as a tree

When solving problems by searching state space, a *destination-state* (or goal-state) is sought from a *source-state* by traversing through a number of *intermediary-states*. Each of these *states* describes the condition of the problem at a particular time along a particular path, whereas the state-space is the whole set of possible states. The term state is sometimes replaced by the term *node*, particularly when the paths through the states are represented as a tree, as shown in Figure 4-1. In that diagram each node, or state, is shown as a circle, with links indicating possible moves from one state to another. It can be seen that in the tree representation of most state-spaces, the number of nodes, at each level is usually larger than the number of nodes in the level above. Following the previous example, the source-state might be the initial positions of pieces at the start of a game of chess. Each link represents one of the possible moves, with intermediate nodes being new positions in the game and the destination-state being an end-game position, perhaps. The solution to a problem, then, is viewed as a series of moves from a starting position. Each new state that can be moved to from a given state is referred to as a *child* of that state. The process of generating the children of a state is referred to as *expanding* that state. If a state has

n children, it is said to have a branching factor of n. If the number of children is n at every state, then the tree is said to have a *branching factor* of n. Expressing this more formally, the total number of paths in a tree with branching factor b and depth d, for example, is b^d. Therefore, the number of paths is said to explode exponentially as the depth of the tree increases. [16]

4.1.2 Search and Heuristics

The purpose of search is to find a path through a state-space that represents a solution to a problem (*e.g.* finding a sequence of moves that lead to a winning end-game position in chess). There may be more than one solution, or destination-state. The best of all possible solutions to a problem is called the *optimum* solution. Search is made more efficient by the introduction of expertise in the form of *heuristics* (a term given to describe a 'rule of the thumb' strategy for moving closer to a desired destination, or goal). Since the state-space may become inordinately large, it is often desirable during search to remove states that do not look promising - this is called *heuristic-pruning*. Strategies that remove states from the state-space are called heuristics. Unfortunately, a path that initially looks unpromising might in fact be one that ultimately leads to the optimum solution to the problem, *i.e.* to the best destination-state. Hence heuristic-pruning may prevent arrival at the best destination state. A heuristic can therefore be seen as a rule of the thumb for dealing with a problem, *i.e.* a usually appropriate method. By its very nature, a heuristic will not always provide the best solution but generally provides an acceptable one; there is a trade-off between finding a good, or best, solution and the time taken to find a solution. During search, each newly encountered state within the state-space is evaluated for its "niceness". The niceness factor that is quantified by a state evaluation function is used to determine the perceived relative success of each possible move. Thus, traversal through the state-space can be governed by the niceness factors of the state. The exact manner of traversal through the state-space is determined by the *Search Algorithm* used (as explained in the following section).

4.2 Traditional Search Algorithms

There are numerous search algorithms in existence ranging from entirely exhaustive algorithms such as the *Depth-First* and *Breadth-First* to the selective *n-tiered, Hill Climbing* and *Branch and Bound* searches. [16] These methods are explained below. It should be noted that many other algorithms exist, but the above listed ones serve to illustrate the fundamentals of search algorithms.

4.2.1 Exhaustive Search

A path to a destination-state represents a sequence of moves in a successful solution to a problem. Viewing all paths to destination-states as being equally good, one simple way to find a successful path is to search exhaustively through the nodes in a tree until a destination node is found. One example of this approach is *depth-first-search*, which is illustrated in Figure 4-2. The source-node is expanded, the next node selected being the left-most of the children. This node is then expanded and again the path to the left most of its children is taken, and so on. If a destination-node is not encountered, then once the tree has been traversed to a specified depth, the path moves back up one layer and the search is continued down from the node to the right of the node previously expanded on that layer. This behaviour can be repeated either until a destination node is found, or until the whole tree has been traversed (indicating that no solution to the problem can be found).

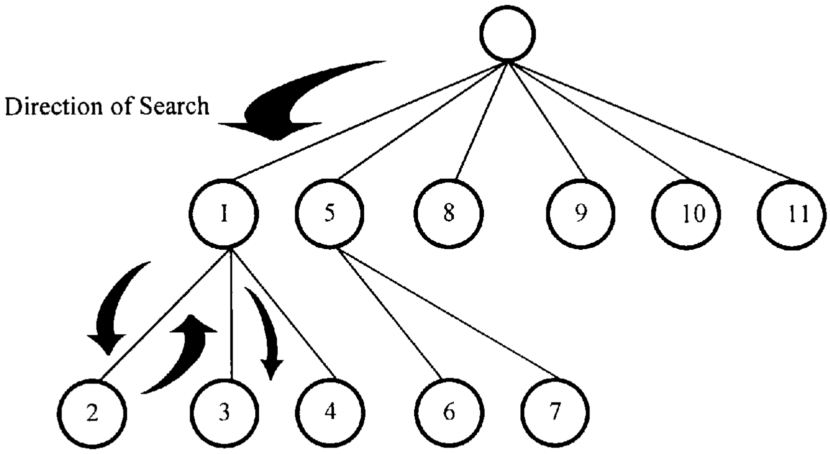


Figure 4-2 Depth first search

Conversely, the *breath-first search* expands all children on a level, in order to locate a destination node, before moving to the next layer down. Expansion here occurs breadth-wise, rather than depth-wise. To express this more exactly, the breadth-first search checks all paths of a given length before moving on to any longer paths (as indicated in Figure 4-3).

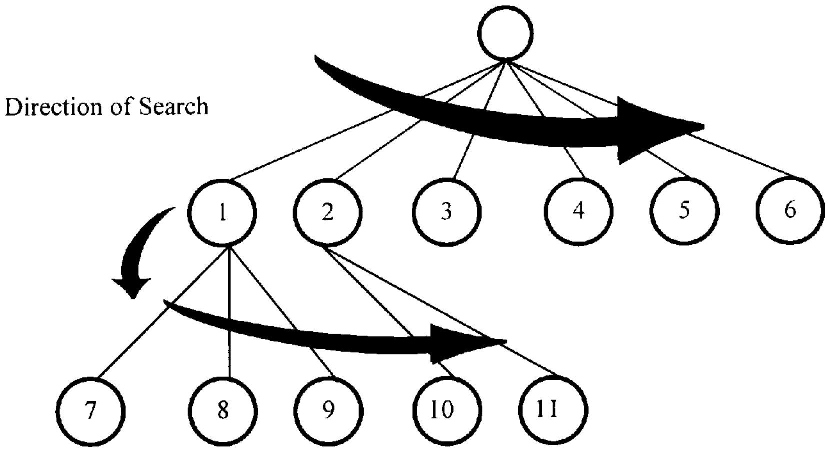


Figure 4-3 Breadth first search

It can be seen then that the depth-first and breadth-first search algorithms take their name from the direction travelled from the source state to the destination state in constructing the tree representing the state space. In Figure 4-2 and Figure 4-3, the direction taken by the search algorithm is indicated by the arrows. The order in which nodes are explored is indicated by the number within each of the nodes. For exhaustive searches in general, the direction - or order - in which states are generated is immaterial - all states can eventually be generated and the best (optimum) destination state selected. However, when the state space is large (as the representations of real-world problems frequently are) this approach may prove costly in terms of time and resources.

4.2.2 Hill Climbing Search

Search efficiency may improve massively when a method for ordering states is introduced. In many problems, a method for measuring the quality of a state can be introduced, *i.e.* a state evaluation function can be employed (as introduced in Section 4.1.2). A search algorithm that uses an evaluation function is said to be *heuristically informed.* Hill-climbing (illustrated in Figure 4-4) is a heuristically informed technique and is a variation on the depth first search algorithm (explained in section 4.2.1). After expanding a state, all children are evaluated using an evaluation function to determine which is considered most likely to lead to a (or the optimum) destination node. Only the 'best' child-node is retained for further expansion - all other children are discarded. This can be important when dealing with a large state-space, as can be seen in Figure 4-4 (in which the number of states discarded is far greater than the number which must be retained; nodes are labelled here by a value indicating their 'niceness').

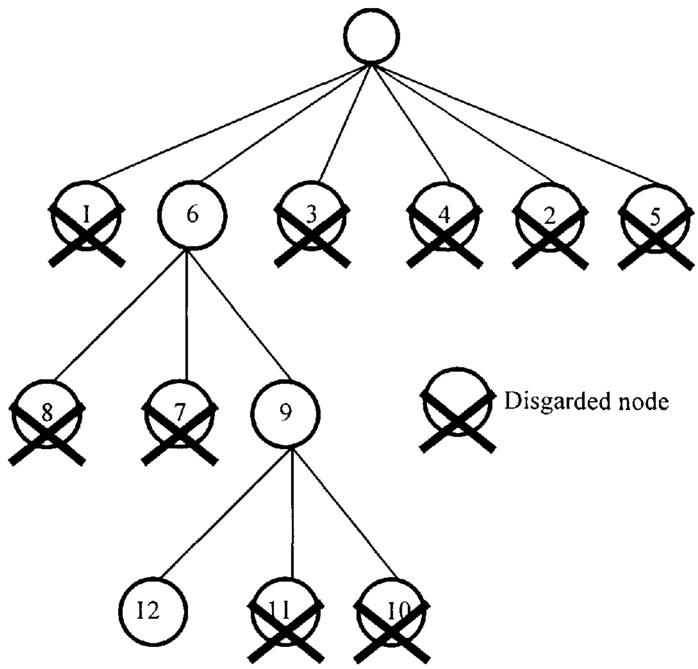


Figure 4-4 Hill climbing search

This approach can often lead to a very good solution in a relatively short time, but some possible destination states are inevitably lost in the process. The state-space is greatly reduced but the method no longer guarantees that the best Destination State will be found. The process of moving from one node to another can be viewed as one of climbing a hill. As the 'best' path is traversed, the value of the nodes being moved to increase towards a destination node, which has a high value and thus may be thought of as a peak. There will be one peak for each solution, with the optimum solution having the highest peak, as indicated in Figure 4-5. This Hill-Climbing search method guarantees to reach one peak quickly, but the exhaustive search methods (explained in section 4.2.1) guarantee to reach the highest peak, albeit in a far greater length of time. Therefore, the main disadvantage of this approach is the reduced likelihood of finding the optimum Destination State.

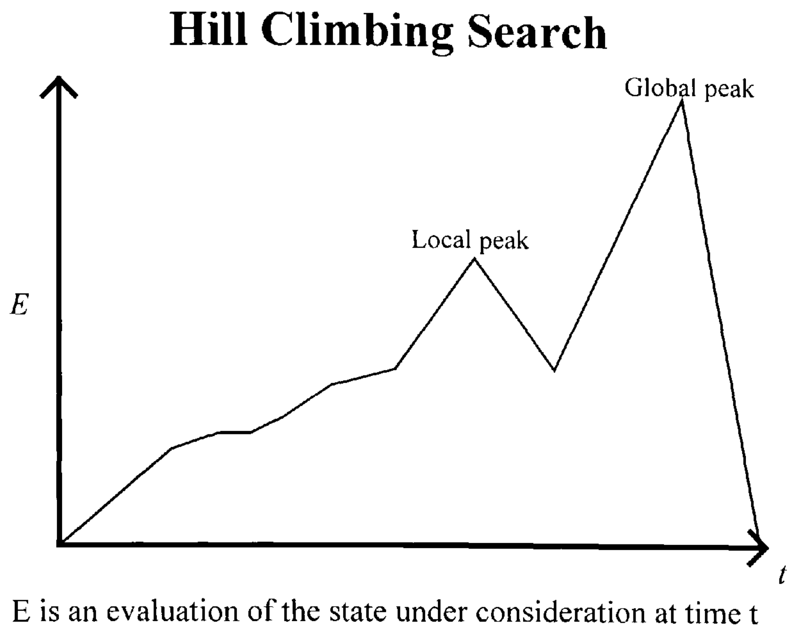


Figure 4-5 Global and Local Peaks

A variant on this Hill-climbing method, and hence another alternative to the exhaustive search is the n-tiered search. This *n* is the number of moves ahead (or layers in the tree) to which the search will progress from a source state before evaluating the niceness. Thus, all nodes *n* moves below a node under consideration will be evaluated before choosing a move from that node. This type of search is common in the field of game playing. [15] Obviously, the larger the number of moves ahead explored, the greater the likelihood that the result returned will be accurate, but the greater the cost in time and resources. In addition, the method has the disadvantage of suffering from the so-called *horizon effect* (illustrated in Figure 4-6).

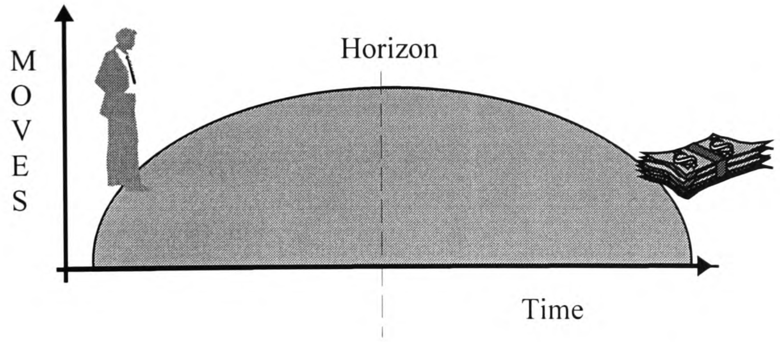


Figure 4-6 The Horizon Effect

The term *horizon* represents the number of moves ahead that the search method examines. In hill-climbing the horizon is set at one move. In n-tiered search, the horizon is pushed further, to *n* moves. However, we still cannot see further than the horizon, and despite the increased likelihood of success being granted by extending the horizon, there is still no guarantee of success.

4.2.3 Branch & Bound Search

The hill-climbing algorithm could not guarantee that the optimum solution to a problem would be found. Where the optimum solution is required, an extension of the hill-climbing algorithm, called the branch & bound search (illustrated in Figure 4-7, where, for example, the shortest distance, by road, between two connected points on a map could be determined), can be used. The Branch & Bound search relies upon some effective measure of how likely it is that a state will lead to the optimum solution, *i.e.* it requires a state evaluation function. Branch and bound is, therefore, another heuristically informed search strategy.

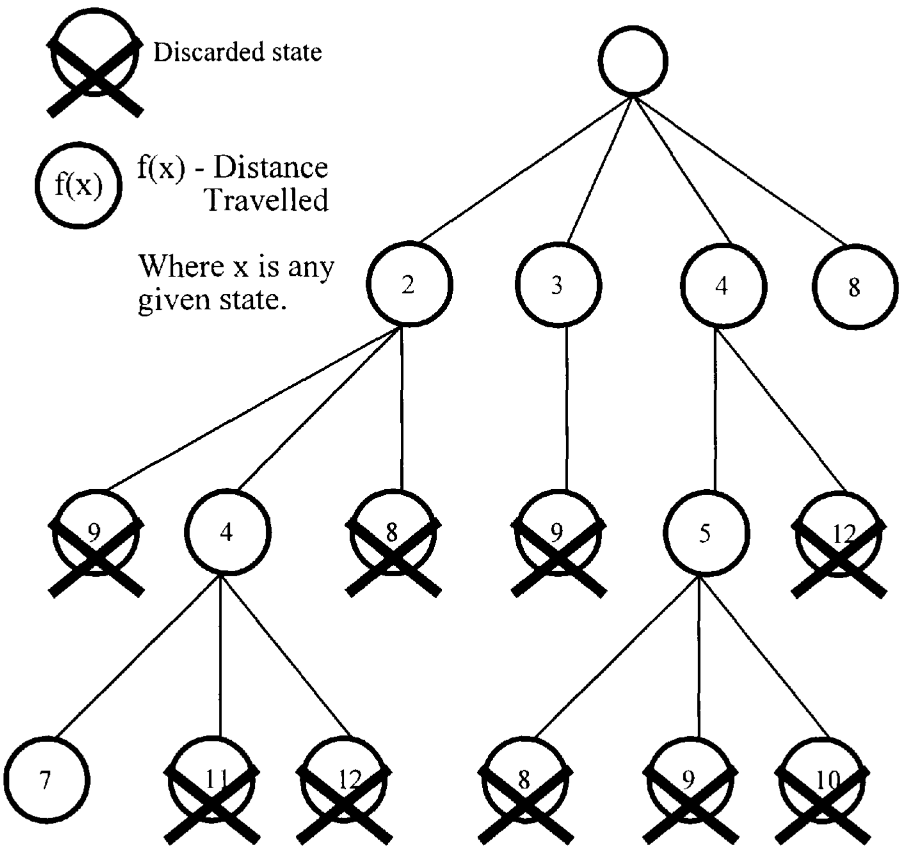


Figure 4-7 Branch & Bound Search

As with the hill-climbing algorithm, when each state is expanded only the best child is immediately considered. The difference is that none of the other children are discarded immediately; instead they are kept for future consideration. A good destination state can be found in a relatively short time, but to find the optimum destination-state, all the remaining states that have been set to one side also have to be expanded. If one of these states anywhere along a path, is less satisfactory than the best destination state found to date then it is discarded.

This method is good when no horizon effect is to be found. Where the measure of effectiveness of a state can vary greatly from move to move, it is not certain whether a sibling state that is currently worse will surpass the best destination state found to date. In these circumstances, removing states that appear worse can result in the optimum destination state being lost (illustrated in Figure 4-8 where, had the node numbered 8 been discarded, the best destination node could not have been reached).

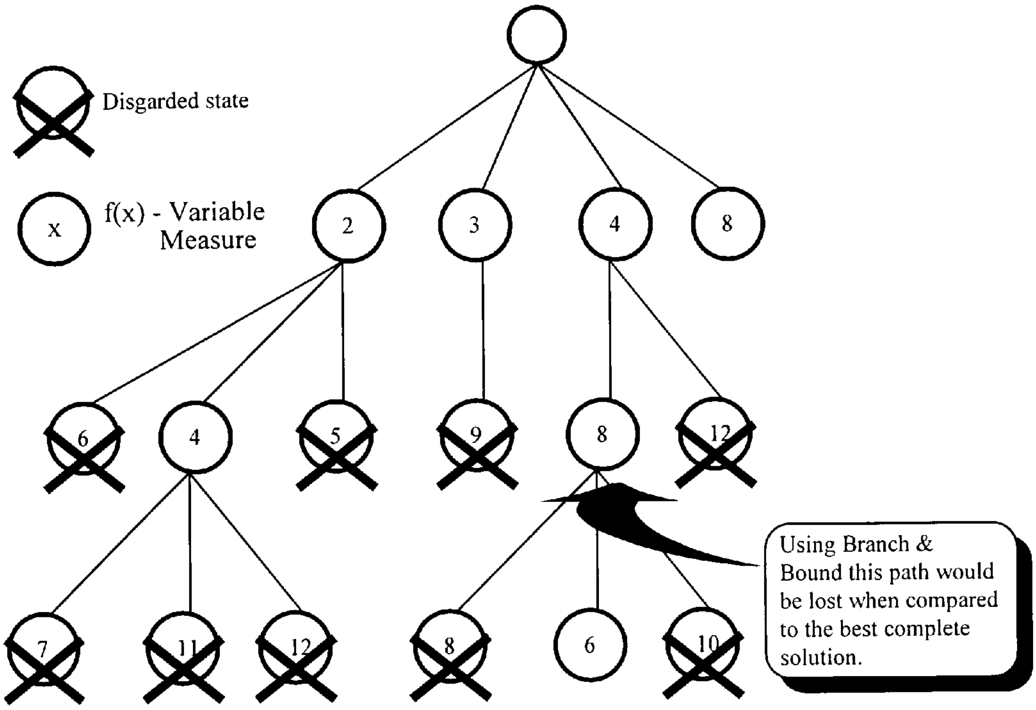


Figure 4-8 Limitations of Branch and Bound Search

4.3 Directed Search

The search algorithms discussed so far are relatively undirected. That is, numerous paths - or alternatives - are generated at each new state. A common alternative search technique is the directed approach. This method of solving problems is normally associated with traditional rule based expert systems. [32, 33, 34]

The following sections outline an alternative problem solving technique, called Directed Search, that is used by AI practitioners as a basis for Rule Based Expert System development. The way in which Expert Systems are developed, how knowledge is represented within Rule Based Expert Systems and how Rule-Bases are navigated are explained. Finally, conclusions are drawn regarding the utility of experts systems.

4.3.1 Introduction to Directed Search

While the terminology of directed search is different from that of undirected search (explained in section 4.2), the underlying strategy is quite similar. With traditional (undirected) search techniques for solving problems, a large number of alternative paths (sequences of moves) from an initial position are considered. Knowledge, in the form of heuristics, is used to *reduce* the number of paths that need to be considered. With directed search, knowledge about the current state of the problem is used to *direct* a single move (although Search forms the foundation for both approaches).

In directed search, *Rule-based* (or *Knowledge-based*) systems use elicited expertise to 'draw inferences' in an attempt to reach a goal. The success of the knowledge-based approach depends entirely on the completeness of the captured expertise. Therefore, the power of the expert system depends more on the quality of its *knowledge-base* than on the strategy used to draw inferences. Difficulties in acquiring and representing the knowledge required to run an expert system has limited their usefulness in solving large problems. Added to the representational and acquisitional problems is the inherent difficulty in determining in which way uncertain or contradictory information should be handled.

4.3.2 Knowledge Representation

Knowledge-based expert systems attempt to emulate human expertise by having a knowledge-engineer first elicit and then instantiate the explicit knowledge of the domain expert in the form of *Production Rules*, better described as premise and conclusion pairs. [33] The main advantages in using production rules to represent knowledge is that once instantiated within the knowledge-base of an expert system they:

- are accessible to the knowledge engineer, allowing the instantiated knowledge to be readily altered, since the knowledge is represented by simple premise-conclusion pairs;
- are accountable to the user of the knowledge-based expert system since all conclusions drawn from the knowledge-base have explicit rules that indicate why an inference was made;
- can be used as a basis for communicating to a user how a conclusion was arrived at, allowing the user to learn more about the domain;

- allow experimentation allowing the domain expert and knowledge engineer to progressively refine the knowledge (or rules) within the knowledge-base.

Production rules are of the form LHS → RHS where LHS (left-hand side) determines the conditions or situations that must be satisfied for the rule to be applicable and RHS (right-hand side) identifies the action(s) that must be taken once the rule is applied.

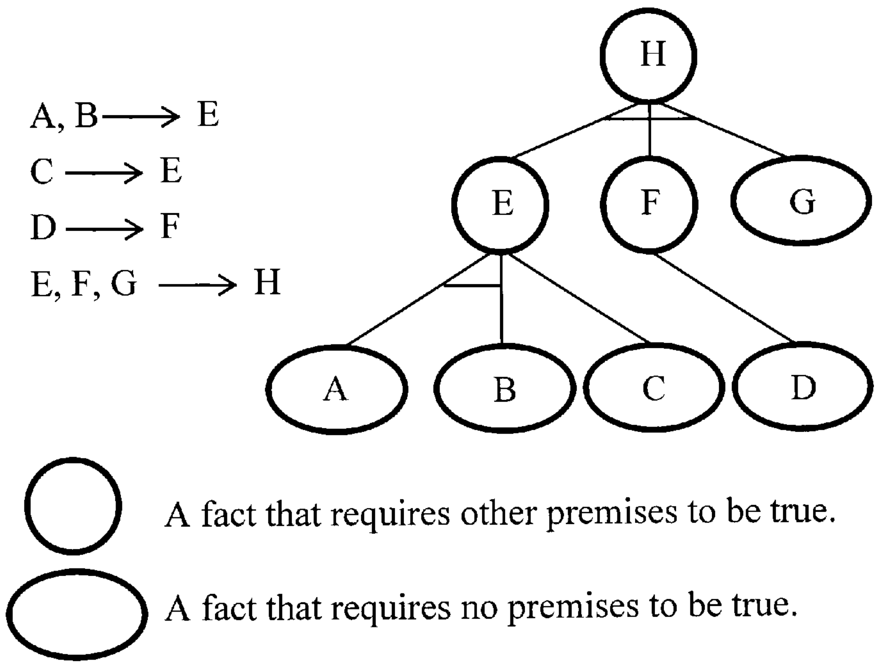


Figure 4-9 Logic Graph

The behaviour of production rule systems can be represented by a *logic graph* (or tree) [33] an example of which is shown in Figure 4-9: the nodes on the graph are facts; the arcs represent the condition-action pairs such that there is an arc from node *M* to a node *N* if there is a rule in which *M* appears on the left-hand side and *N* appears on the right-hand side; leaf nodes (nodes at the end of branches) are facts that require no other premises to be true. When the left-hand side of a rule consists of several conditions C1, C2, ... Cn, then the corresponding arcs are joined to emphasise the conjunction (in Figure 4-9, A and B together imply E, as opposed to C alone implying E).

The problem is, essentially, still one of applying search in an attempt to reach a goal. A sequence of production rule applications must be found that represent the facts which support, or solve, the goal. The goal corresponds to the concept of the Destination State introduced in Section 4.1.1. The direction the search takes through the AND/OR Graph - either backwards or forwards - is determined by the type of problem being solved. *Backward Chaining* and *Forward Chaining* are the names given to the direction the search takes through the graph and explained in the following section.

4.3.3 Inferencing Strategies

This section introduces the two methods, Backward Chaining and Forward Chaining, of navigating a rule-based system. The method selected by the Knowledge-engineer is problem dependent. This section is intended to provide the reader with a keener insight into problem solving by using AI, and in particular of how search is the foundation of all such approaches.

4.3.3.1 Backward Chaining

Backward Chaining - or *Goal Based Reasoning* - begins with a conclusion for which a proof is then sought. In other words, the approach begins with the goal of the problem, and works backwards in the hope of finding known facts within the knowledge-base; these facts, and the (reverse) sequence of rule applications, constitute a solution to the problem. This approach is useful where only a small sub-set of the available facts is required to reach a solution.

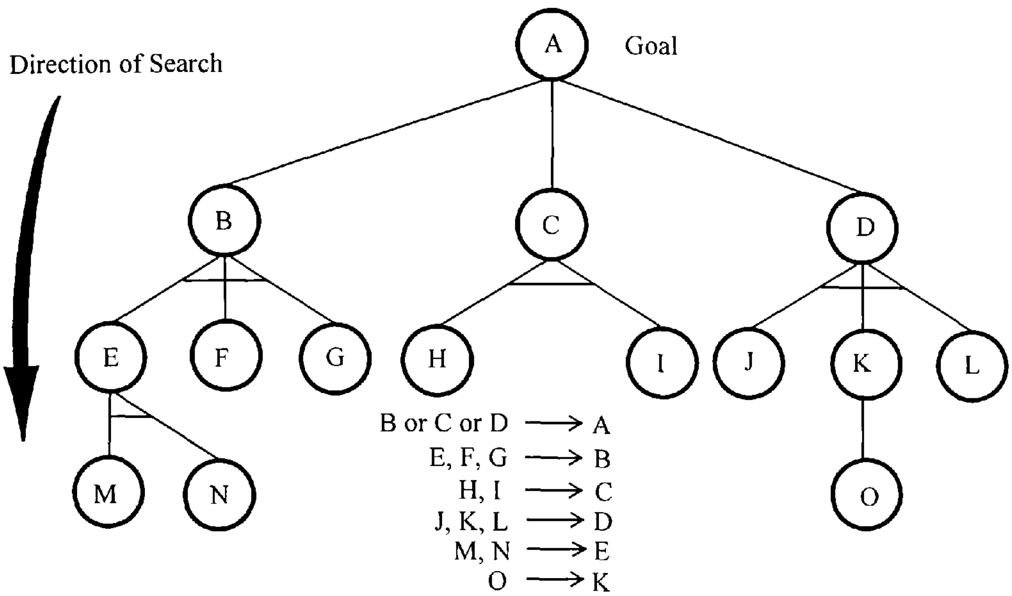


Figure 4-10 Backward Chaining

The example of backward-chaining given in Figure 4-10 shows an attempt to prove goal A. The rules for the conclusion A are B, C or D (as can be seen from the production rule B or C or D→A). Therefore, for A to be true, the truth of B or C or D must be determined. By using a Depth-first search (see Section 4.2.1), establishing the truth of B, C and D becomes a relatively straightforward task. The premises for the conclusion B are E, F and G (as can be seen from the production rule E,F,G→B). Therefore, for B to be true, the truth of E, F and G must be determined. Figure 4-11 indicates the order of the search by associating a number with each node. (In this example, it is assumed that each path down the layers of nodes ends in the successful establishment of truth by encountering a known fact.)

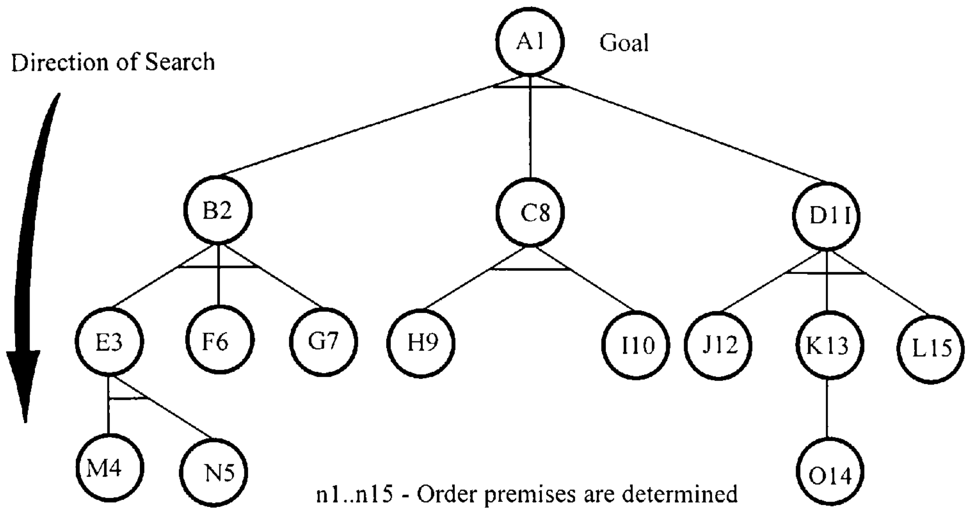


Figure 4-11 Depth-first Backward Chaining

Although not critical to solving a problem, some importance can be placed upon the ordering of the rules. An example of when this might be important is in a very large rule set where *consultation periods* (time spent by the user navigating the knowledge-base attempting to find a solution) could be shortened by ordering the set more efficiently. Where a goal can be proved true by a variety of different paths through the graph (that either domain expertise or experience using the knowledge-base reveals) it would be sensible to order the rule set in such a way as the shortest path is taken; Figure 4-12 demonstrates a re-ordering of rules in Figure 4-10 which allows A to be proved more quickly by the straightforward proving of C, rather than by first attempting to prove the more involved B. Alternatively, it may be the case that proving B is the most common solution to the problem and, although processing the rule is more involved, it is more sensible to process B before other, more simple, rules.

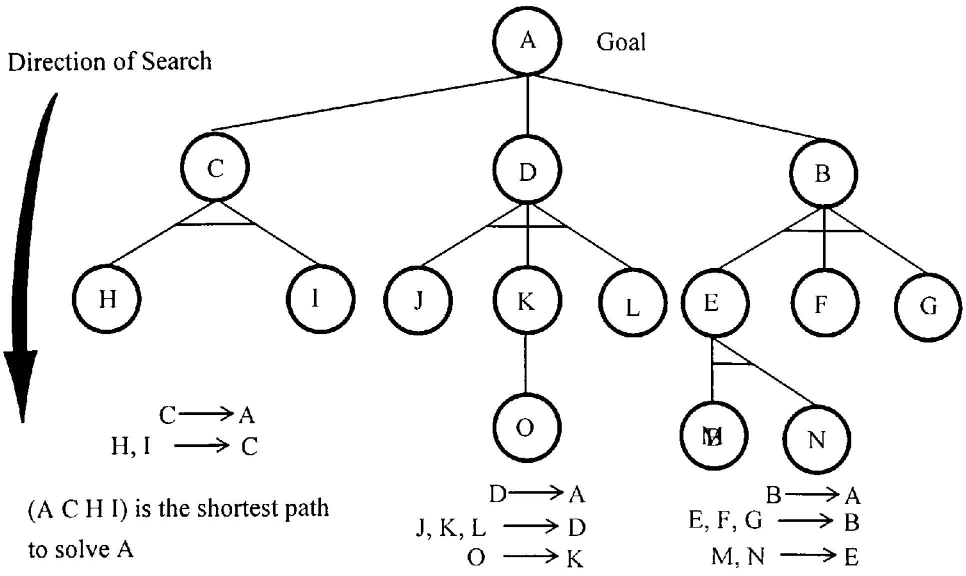


Figure 4-12 Depth-first Backward Chaining with Ordered Rule-set

4.3.3.2 Forward Chaining

Forward Chaining - or *Data-driven Reasoning* - (this approach begins with known facts and uses them to attempt to find a path that reaches a destination goal) is a procedure for making inferences that is useful when a large amount of data are available from the outset (see Figure 4-13). Facts are matched against available rules to infer new facts on a path that reaches a destination goal. Since many rules may use the facts available to the system, the actual choice of rule at each stage is determined by the use of heuristics called *Conflict Resolution Strategies.* [16]

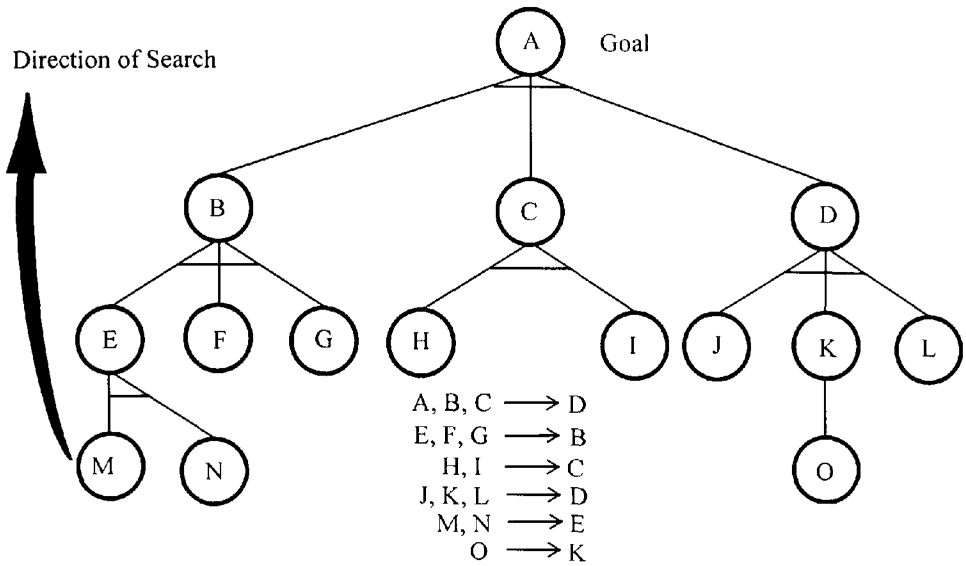


Figure 4-13 Forward Chaining

All rules that have facts available to them are collected in a conflict set. Where there is more than one rule in the conflict set - or more than one available path from which to choose - some method must be introduced to determine which rule to execute. This method is called *the conflict resolution mechanism.* Most conflict resolution strategies are simple. Often a combination of these strategies is used to ensure elimination of all possible conflicts.

Although introduced here within the context of conflict resolution in a forward-chaining inference engine, deciding which order a particular rule should be considered can have far reaching consequences within search generally; Section 5.2.2 contains an explicit example of how altering the order in which rules are considered permits alternative solutions to be generated. Some examples of conflict resolution strategies within inference engines follow [33]:

- *Elimination of executed rules:* Under this strategy, instantiations of the rule most recently executed are discarded from the conflict set. This ensures that a rule is not processed more than once on the same set of facts. Therefore, undesirable loops in the path to a goal can be avoided. This strategy is also called *refractoriness*. [34]
- *Textual position:* A rule placed earlier in the rule set is selected first. This approach is useful when the knowledge engineer has ordered the rule set with the most important - or most often used - rules first.
- *Rule prioritising:* Instead of ordering the rule set, rules are assigned a priority that governs their importance.
- *Specificity:* The rule in the conflict set that uses the most facts - or has the most premises - is selected. This assumes that the rule with the most premises is the most specific to the problem.
- *Recency:* Dependant upon the choice of recency - least recently or most recently applied rule - priority is given to the rule accordingly. Where the most recently used rule is selected, this approach resembles a depth-first search. Alternatively, where the least recent rule is selected the approach resembles a breadth-first search. [33]
- *Data prioritising:* Here premises are assigned priorities. Since premises may appear in more than one rule it is sometimes possible to rank their importance.
- *Random choice:* If after applying a number of different resolution strategies several rules still remain in the conflict resolution set, one is selected at random.

Since rules can be added to the knowledge base (rule-set) incrementally, the development of rule-based expert systems lends itself to the rapid prototyping design methodology. [1]

4.3.4 Difficulties with knowledge elicitation

In fields where explicit knowledge is difficult to come by (such as the stowage planning problem where experts knowledge is implicit), an approach to problem solving that is entirely based on production rules would not be feasible. This is primarily due to the difficulty in obtaining the required knowledge from domain experts. This difficulty can arise from any combination of the following reasons:

- an expert may deliberately resist questions due to personal reasons (such as a feeling of vulnerability);
- an expert may not be able to articulate expertise due to problems explaining complicated facts;
- finally, the expertise in question may be hard to represent in a symbolic manner due to it being largely intuitive.

4.4 Expert System Development Cycle

This section outlines the development of the Expert Systems in order that the work detailed in Chapter 5 and the approach taken by this author, detailed in Chapters 6, 7 and 8, be understood. It should be understood that, although the term Expert System is often attributed to rule-based systems, the author uses it to describe any system that performs the role of an expert in a domain.

Stage	Description
Identification	Determining problem characteristics.
Conceptualisation	Finding concepts to represent knowledge.
Formalisation	Designing structures to organise knowledge.
Implementation	Formalising rules and heuristics that embody knowledge.
Testing	Validating rules and heuristics that embody knowledge.
Revision	Redesigning and refining the system.

Table 4-1 Rapid prototyping expert system development life cycle

The development of an expert system is evolutionary and, except for very small clearly defined tasks, the approach taken is impossible to strictly regiment. [33] Rather, expert system development lends itself to rapid prototyping. The system should grow incrementally with new and better strategies being adopted as they become apparent along the project life cycle. [Ibid.]

Stage	Description
Demonstration prototype	The system solves a portion of the problem undertaken, suggesting that the approach is viable and system development is achievable.
Research prototype	The system displays credible performance on the entire problem but may be fragile due to incomplete testing and revision.
Field prototype	The system displays good performance with adequate reliability and has been revised based on extensive testing in the user environment.
Production model	The system exhibits high quality, reliable, fast and efficient performance in the user environment.
Commercial system	The system is a production model being used on a regular commercial basis.

Table 4-2 Expert System staged development

An expert system is composed of a knowledge base, which is used to make inferences (using an *Inference-engine*). The traditional knowledge engineering approach follows the expert/knowledge-engineer/knowledge base paradigm. According to this paradigm, a knowledge base is created by knowledge engineers who extract knowledge concerning a specific area (or domain knowledge) from one or more experts. These knowledge engineers then translate this elicited information into a format suitable for the inference engine. Experts then conduct trial consultations to verify that the knowledge base is correct and complete. The development of an expert system takes the form of a multiple feedback loop (see Table 4.1) with each stage of the process being iterated until a satisfactory implementation is produced. [35] An expert system can be developed incrementally, that is to say that the degree of completeness of each implementation can be increased progressively. A possible staged implementation procedure is given in Table 4-2.

4.5 Conclusion

Search is the foundation upon which AI rests, whether it be searching state-space or consulting a set of rules. The search techniques mentioned above should be seen as a sub-set of tools from which an AI practitioner can draw in order to solve intelligent problems. The choice of tool can be problem specific where only a limited number of approaches may be applicable. Reference will be made in subsequent chapters to the techniques described in this chapter.

5 EVALUATION OF RELATED WORK

5.1 Introduction

The search for an efficient procedure for container ship stowage planning has drawn the attention of shipping companies and academic researchers since the 1970s. The main aim of such a procedure can be defined as minimising the overall cost of the shipping operation subject to a set of given constraints. Three areas of research into how computers can aid the shipping process have arisen, namely:

- automating container-terminal operations;
- automation of shipping operator operations;
- and providing computer tools that assist personnel perform container-terminal and ship stowage planning tasks.

This thesis deals specifically with the automation of container-ship stowage planning. The methods used for producing solutions for the stowage planning problem have been grouped into the following main classes: simulation based upon probability, heuristic driven, mathematical modelling, decision support systems and rule-based expert systems. [36] None of these approaches has, to date, provided an optimum - or best -solution to the problem. This chapter outlines and critically analyses work already published in the area of computer tools for ship stowage planning examining one characteristic example of each approach in turn. The relative strengths and weaknesses of each approach are highlighted, and are referred to in subsequent chapters to explain the motivations for the approach taken in this research project.

5.2 Simulation based upon probability

The Monte Carlo simulation method adopted by J. J. Shields in his Computer Aided Pre-Planning System (CAPS) constitutes this class of approaches. [13]

5.2.1 Introduction

The method considered in this section is an approach to container stowage optimisation due to Shields. [Ibid.] This system, referred to by Shields as the Computer Aided Pre-Planning System (CAPS), is a package comprised of several software modules. These modules have been designed to aid the planner with the many stages of the entire planning process. Of particular interest in the review that follows are the modules responsible for providing the planner with a number of possible stowage solutions, from which the planner may choose a preferred solution. Shields uses probabilistic information to produce these alternative solutions. The following section outlines the work with container stowage optimisation completed and implemented by Shields.

5.2.2 The Computer Aided Pre-planning System

CAPS is a collection of software modules that support all stages of the process of planning containerised cargo stowage in the journey of a vessel. CAPS produces a number of possible stowage solutions for every port along the journey based upon simulation of the whole route and then presents the most cost effective stowage pattern found. The system modules which support this process are:

- A *stability module* that permits container-ship stowage data to be input and displayed along with the consequences that particular stowage has on vessel stability. Data are entered by the user and the resulting changes to heel, draft (explained in Section 2.3.5.2), GM (the distance between the vessel's centre of gravity and its metacentric height, explained in Section 2.3.5.3), trim (explained in Section 2.3.5.4), stress (explained in Section 2.3.6), and dead-weight (the difference between the loaded and unloaded weights of the ship) are updated automatically.

- A *statistics module* that produces a variety of reports and provides forecast information on anticipated cargo at subsequent ports, for the stowage (pre-planning) module, below. This module requires information from a database of stowage data.

- A *port module* that contains information about each port facility. This module uses a database containing information about draft restrictions (explained in Section 2.3.5.2), berth availability (explained in Section 2.3), arrival time limitations (limitations upon when a ship can be berthed), crane specifics (explained in Section 2.4.1) and working hours (limitations upon when, and for how long, Stevedores can work).

- A *fuel module* that records and forecasts fuel consumption during each voyage.

- A *stowage module* which automates the pre-planning process in two distinct parts: the generation of possible ship loading patterns; and the evaluation and ranking of these loading patterns according to cost efficiencies. Shields claims that the algorithm used closely models the thinking process of the human pre-planner. A full discussion of this module is presented in the next section.

5.2.2.1 Description of the pre-planning algorithm

Containers are allocated to stowage positions in groups. Each group consists of containers with the same characteristics. For example, one group could be 40' refrigerated containers destined for Hong Kong. The groups are processed by the algorithm one at a time, beginning with the group destined for the furthest port in the set of containers to be loaded, working backwards to the group destined for the nearest port.

The first step, when processing a container grouping, is to create a set of all the legal positions where the group could be placed. Here, a legal single stowage location is defined as a slot that is unoccupied and will accept a specified container without violation of stowage constraints (see Section 2.1.3.3 for examples of violations). Once the set of available slots for a group has been defined, the algorithm then searches for the optimum placement within this set for each container in the group. The set of mappings of containers to slots is progressively pruned according to criteria that, Shields claims, a planner would keep in mind when performing the task of container placement manually. Examples of the criteria checked are:

- to avoid overstows;

- to load heavy containers low in the ship;
- to stow containers with similar characteristics in the same hold.

Each of the stowage criteria is considered in turn, and the members of the current set of legal positions that fail that particular criterion are removed. After a criterion has been processed and the set of legal positions pruned, the remaining members of the set are checked to see if they all fall within the same transverse row of stacks of the ship. If they do, then the containers from the group are allocated to those positions. Searching for a group of legal positions in the same transverse row will generate better block stowage than would otherwise be the case. If they do not, then the next criterion is applied. When all the stowage criteria are exhausted without a unique row being found, a random row is selected from the remaining members of the set.

In the above scheme, the resulting stowage solution depends upon the order in which the stowage criteria are considered. Very often, stowage criteria will contradict each other. For example, stowing heavy containers low in the ship will result in over-stowage since the criteria here for ordering containers in a stack may result in heavier containers with a closer destination being stowed under lighter containers with a destination further away. Shields addresses this by assigning a probability for each of the stowage criteria with the assigned value being relative to the importance of the consideration. Instead of simply selecting the next stowage criterion in sequence, the system randomly proceeds through the stowage criteria. Given that probabilities are assigned in order of relative importance this random process has the effect of generally, though not always, considering more important considerations first. The user can assign different probabilities to the stowage criteria and in this way generate a hierarchy that is random, but biased toward a particular arrangement. (This randomness constitutes the so-called Monte-Carlo aspect of Shields method.)

Once all containers have been allocated a slot, the resulting stowage pattern is evaluated and a *penalty score* allocated to it. The penalty score reflects various operational costs that are incurred as a result of the stowage pattern. Shields has defined eleven categories for which penalties can be determined:

- *Over-stowage* -- This penalty reflects the cost of re-handling containers within a port, and takes into account the varying costs associated with different ports.
- *Hatch access* -- Excessive hatch access is a symptom of poor block stowage of cargo having the same destination.
- *Port restriction violations* -- Very often ports are unable to access some slots due to physical restrictions associated with the berth.
- *Cargo left behind* -- Failing to load all cargo is highly undesirable.
- *Stowage over void spaces* -- This reflects on-deck stowage above unused below-deck slots and has a loss of revenue associated with it.
- *Lashing penalties* -- This reflects labour cost associated with securing on-deck containers.
- *Incomplete transverse rows* -- This reflects additional longitudinal crane movement to fill rows left only partially full at previous ports.
- *Mixing of lengths* -- This penalty reflects the loss of crane productivity associated with having to handle variable length containers in the same transverse row.

- *New destination cargo added to rows* -- This penalty is assigned each time a container is loaded into a row that previously did not have other containers of the same destination. High penalties associated with this category are an indication of poor block stowage.
- *Ballast required* -- This penalty reflects the fact that carrying sea-water is wasteful.
- *Stability penalties* -- These penalties are high and are associated with GM, trim, draft, heeling or stress constraints that are not satisfied.

5.2.2.2 Producing a solution that takes subsequent ports into account

The algorithm described in Section 5.2.2.1 is repeated to generate a number of solutions, each having a penalty score, from which the user may choose. This yields a set of alternate solutions to choose from. This is possible because the randomness in ordering the criteria for making container placements allows numerous stowage patterns to be generated. However, these criteria only consider factors at a single port. Shields rightly points out that this, in itself, is insufficient in producing effective solutions due to the multi-port nature of the problem. Rather, containers to be loaded and discharged at further ports are considered by simulating the voyage of the vessel through further ports, repeating the process described above by using an increasing amount of statistical forecast information about containers. The penalty scores associated with each of the paths generated, through subsequent ports, are compared to find the best path.

5.2.2.3 Results obtained by CAPS

American President Lines state benefits[13] by the use of CAPS, which include:

- Increased vessel capacity of as much as ten percent or 250 TEU's.
- More precise vessel-to-container allocation allowing better fleet utilisation.
- Some reduction in over-stowage, although this has not been quantified.
- Fuel oil savings brought about by improved trim (and presumably, although it is not stated, reduced ballast).
- The historical database associated with the application has proved useful for decision making by planners.
- Facilitating the pre-planning process enabling planners to use their time more effectively.

5.2.3 Observations

Shields [*ibid.*] fails to provide information about how out of gauge containers and other special cargo is dealt with, nor how hazardous cargo placement restrictions are dealt with. Such considerations are not even alluded to. (This may be due to the fact that CAPS was developed for an American shipping line; less special cargo may be present on American routes.) [12] The name of the system (CAPS - *Computer Aided* Pre-planning System) strongly indicates that the system is intended only as an aid, albeit a broadly useful one, to the planning process; no claim is made that <u>all</u> aspects of the planning process are automated (only that they are supported). It seems reasonable to assume that the stowage plans may require subsequent alterations by the planner to take into account complicated omitted factors such as special cargo.

The benefits stated by Shields culminate in the statement that pre-planning has been facilitated by the implementation of the CAPS system. Although it is indicated that some reduction in over-stowage has been realised it is unclear whether this is a direct result of the effectiveness of the stowage module, or merely the result of a general trend in the container transport industry of adopting stowage tools that allow the relatively rapid conceptualisation of stowage patterns. Other shipping companies that have either replaced or augmented paper-based stowage planning systems with computerised methods have experienced a similar improvement in cost effectiveness. [12] The benefits produced by CAPS could, therefore, be mainly due to the computerised facilitation of other parts of the planning process, rather than necessarily being attributable only to the stowage module.

Whereas the stowage algorithm described above is certainly novel, it is difficult to envisage that the approach generates anything close to an optimal solution, due both to the simplification of the problem (omitting special cargo) and to the limited number of alternative solutions explored. Given the relatively limited nature of computer hardware at the time of the development of CAPS (fifteen years ago) an approach that generates only a limited number of valid solutions to a simplified version of the stowage problem is understandable. The approach adopted assigns containers to stowage locations on a one to one basis and given this basis Shields favouring of a random simulation procedure in preference to the usual analytical tools for solution of combinatorial optimisation problems, such as linear programming (see Section 5.4), is a valid one.

5.3 Heuristic driven approaches

This second class of automated planning processes incorporates human planners experience encoded in the form of heuristics. [4] These heuristics can produce a complete, but rarely optimum, solution without the interaction of a user. This class of approaches is also employed in work on *packing* (described in Chapter 8.3.2) that is related to this kind of problem. [37, 38, 39, 40, 41, 42]

5.3.1 Computerised ship load sequence planning at a terminal

The project described in this thesis is concerned with the container-ship loading process from the point of view of the shipping operator planner. The work described in this section, due to Martin *et al* [4], is concerned with the container-ship loading process from the point of view of the container-terminal planner. Specifically their program takes a ship-operator planner's stowage-plan and produces a loading sequence to be followed at the container-terminal. Despite the difference in aim of the two projects, an analysis of the work of Martin *et al* is useful for two reasons: it demonstrates the use of heuristics in stowage planning; it clarifies the role and practices of the terminal-planner. The program due to Martin *et al* is designed to serve a container-terminal which uses gantry cranes, transtainers, (described in Section 2.4.1) and trucks to handle containers.

The method recognises the following constraints:

- ship stability;
- the placement of containers in a bay according to length;
- limits on stack height in under-deck bays;
- limits on stack weight in on-deck bays;
- placement of refrigerated containers.

Provision is made for the user to rearrange over-stowed containers and place out-of-gauge containers. For this reason the method produces computer-assisted rather than entirely computer-generated loading plans. The purpose of the method is to minimise transtainer movement and re-handles (containers manipulated more than once during the load process described in Section 3.2.2) within the container-yard. Martin *et al* use a heuristic search (introduced in Section 4.1.2) method to produce solutions. Further, they claim that the model uses strategies similar to those of a human planner.

5.3.2 The approach taken

5.3.2.1 Assumptions underlying the approach

The model of ship planning that Martin *et al* used to develop their approach was based on observations of the Port of Portland's Terminal 6 on the Columbia river. This container-terminal receives containers by road, barge or rail, and is designed, primarily, to handle 20' and 40' length containers. A study of the cargo types typically handled at the port prompted the authors to simplify their model of activity at that port.

The observations they made were:

- that empty containers are usually handled by fork-lift trucks, not by transtainers;
- that very few containers handled were of a hazardous nature;
- similarly, very few containers were out-of-gauge;
- that transtainer-yard type loading equipment for non-empty containers at the port consisted of a combination of transtainers, trucks and cranes;
- newer, dedicated container-carriers have sophisticated ballast tanks that permit stability to be adjusted during, and after, loading, this technology simplifies stability considerations during loading;
- all containers have arrived and been allocated yard slots prior to the load planning process;
- generally, not all of the trucks in the container-terminal are used at the same time;
- all containers in a yard are grouped together according to port of destination, length and weight.

The above observations were used to construct a set of assumptions concerning conditions at a port, on which Martin *et al* based their program. These assumptions are summarised in

Table 5-1.

1. Hazardous and oversize cargo can be excluded, as can empty containers.
2. Containers will be handled by transtainer-truck-crane; no other transport need be considered.
3. Material handling financial costs will be considered as more important than the financial cost of ensuring ship stability by the use of ballast.
4. All containers for a voyage have arrived and are in the yard prior to the planning process, and are generally grouped together according to port, length and weight.
5. Transtainer and cranes will be paired (one to one) and a ship can always be serviced by one or two cranes simultaneously.
6. There will be a surplus of trucks for transporting containers within the container-terminal.

Table 5-1 Container load-planning assumptions

5.3.2.2 Prioritising the factors which affect the time of loading

Martin *et al* report that a study of the three step process, shown in Table 5-2, of loading containers onto container-ships revealed that transtainers effectively dictate the speed of the loading process.

1. A transtainer moves a container from its yard slot to a truck chassis.
2. The truck transports the container to the crane.
3. Finally, the crane loads the container onto the container-ship.

Table 5-2 Container load sequencing

However, although transtainers are usually the limiting factor (in this port), in certain circumstance the time taken for the crane to load a container can be even more critical. An example of when cranes may become a limiting factor in the loading process is when they move excessively between bays. Excessive crane movement between bays occurs when the sequence in which containers arrive at the berth is poorly managed. An analysis of the factors involved in the container loading sequence revealed an inherent prioritisation within the process, shown in Table 5-3.

1. Minimise changes by the transtainer to non-adjacent sections.
2. Minimise transtainer re-handles.
3. Minimise transtainer movements and total move distance.
4. Minimise crane movements.

Table 5-3 Container load-planning priorities

Martin *et al* uses this prioritisation in their program, which enhances transtainer productivity before any other consideration. Transtainer efficiency is maximised by minimising the distance travelled and the number of containers handled. Crane efficiency is maximised by ordering the arrival of containers at the berth so that the distance travelled by the crane is kept to a minimum.

5.3.2.3 The nearest container heuristic

Martin *et al* uses a heuristic search method to minimise the operating costs incurred by a container-terminal during the loading and unloading of container-ships. Martin *et al* give as the measure of success of their heuristic that their program should be able to load ninety percent of the target population of containers, ninety percent of the time. Their load-planning heuristic was developed by considering, initially, the load planner's *sequence plan* (a plan outlining the order that stowage locations on the container-ship are to be filled). This sequence plan contains the following information:

- the order of destination port in which containers will be filled within a bay;
- targets for the number of light, medium and heavy containers for each bay and port combination (*i.e.* for each set of containers destined for the same port which are to be placed in the same bay);
- the crane to be used for each bay and port combination when more than one crane is to be used.

Each bay is filled bottom-to-top, tier by tier, riverside to berth-side (that is, the cells furthest away from the crane first). Containers, within each bay-port combination, are selected that meet the cell target type (taken from the shipping companies stowage-plan). However, target weights on the shipping companies general stowage plan are not considered important since this would only cause instability problems that could be corrected later. When an on-deck stack is close to its maximum weight limit care is taken to prevent additional placements resulting in an in-feasible solution. Similarly, when an under-deck stack is close to reaching its height limit care is taken not to exceed this limit. In addition to weight and height, two other factors considered when loading are destination port and length.

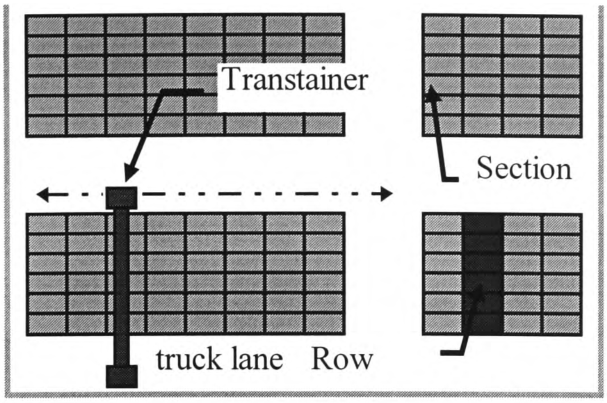


Figure 5-1 Top view of container yard serviced by transtainers

Once the class (determined by destination, weight and length) of the container to be loaded has been determined, the search heuristic attempts to locate a suitable container that meets the load-planning criteria identified in Section 5.3.2.2. The search process begins at the row of containers where the transtainer is currently located and identifies all the containers in that row that match the requirements. If more than one container matches the requirements, the one that minimises material handling costs is selected for transportation to and loading on the container-ship. If a container of the required type does not occupy the row covered by the transtainer, then the search is expanded to include rows within the current section (see Figure 5-1 for an illustration of how a container yard is divided into sections and rows that a transtainer can traverse easily). This process requires good container-yard management to be effective.

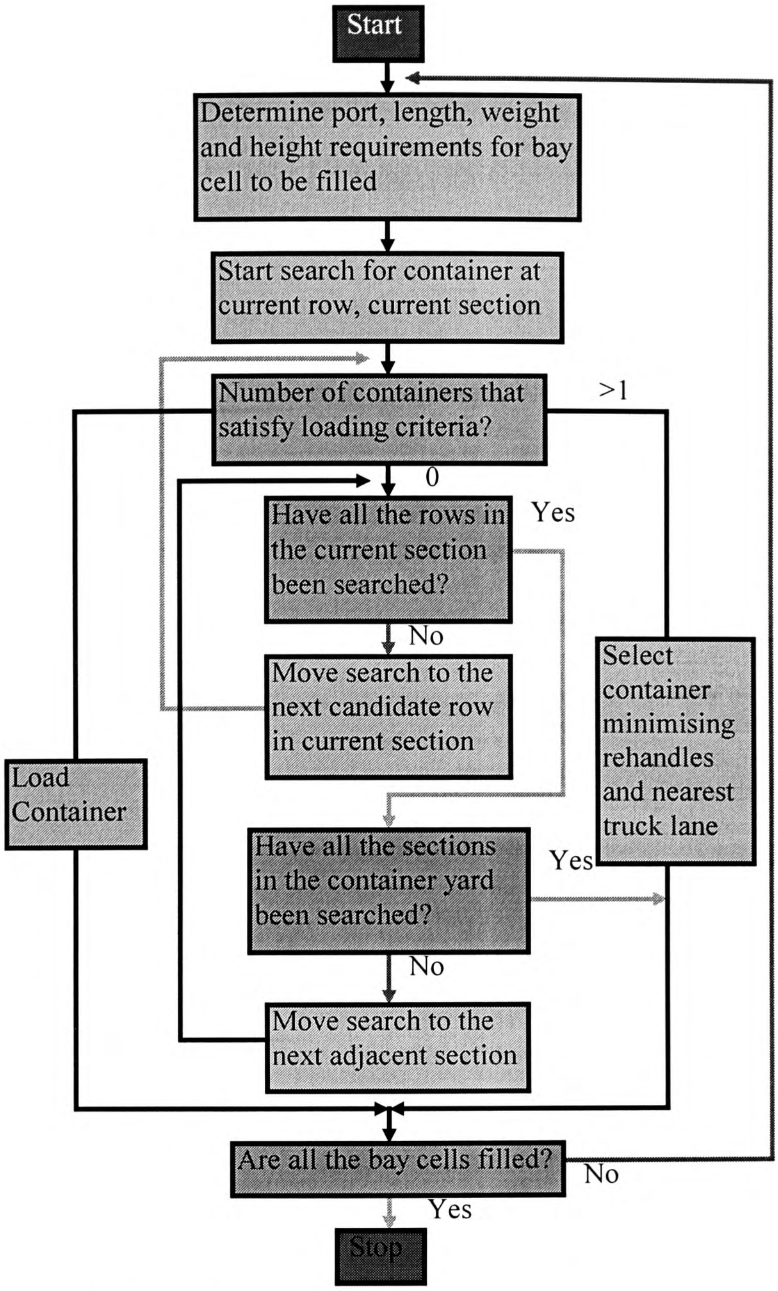


Figure 5-2 Load-planning heuristic flowchart

Once all rows within a section have been searched, the container that minimises material handling costs is selected. If the container of the required type can not be found within the current section then sections adjacent to the current location are added and considered. If an acceptable container can not be found then the search returns to the original section (since this is where the transtainer is physically located). At this point, the required weight for a candidate row to be considered is altered so that any weight is acceptable. By making all weights acceptable then any row not already checked would become a candidate row. The search process is repeated and a container is selected from the match set that reduces material handling costs. If a container, still, can not be located, this event is noted and the search process begun again with the next bay stowage location, now, being considered (see Figure 5-2 for a summary of this loading process).

When the heuristic is attempting to find a container to place at the top of an on-deck stack, then special consideration is given to ensure that stack weight limits are not exceeded. In this case all containers whose weight comes closest to the stack limit are considered with the one that minimises material handling costs being selected. In this way, the lightest containers are dealt with last. Special consideration is also given to refrigerated cargo (Reefers). Reefers are dealt with in the same way as other cargo by being treated as a distinct bay-destination grouping.

Once the vessel has been loaded, stability is then calculated. Two measures are used, that of metacentric height (GM) and trim. The shipping operator assigns values for both GM and trim that must not be violated. A GM value is required as if the centre of gravity is too high the ship may capsize when it rolls and, conversely, if the centre of gravity is too low then the ship may right itself too quickly causing cargo to come loose. A trim value is required as it is desirable to ensure that the vessel's bow is not lower than the stern so that slamming forces are reduced and propeller immersion maintained.

5.3.2.4 Evaluation of the program

5.3.2.4.1 The method of evaluation of the program

The program was evaluated by considering load sequences obtained for two actual voyages each of two separate container-ships. The results and costs of the actual load plans produced by human planners were compared with the results and costs which would have been incurred by the load plans produced by the program. Information about containers to be loaded and the condition of the container-ship when it arrived in port, along with the assignment of ports to cells (the decision that certain cells would be used for containers of a specified destination port) for the voyage and the sequence for loading the bays was provided by the container-terminal.

The program due to Martin *et al* was used to generate a number of alternative load-plans by varying the row and container weight ranges (explained in Section 5.3.2.3). Of the four vessel loadings, one vessel loading was not complete; two containers were left on the berth. This occurred because *light* containers were loaded early in the process and were not available for loading onto on-deck bays later. This in turn resulted in deck stress limits being exceeded for two stacks. This problem had to be corrected manually by rearranging six containers. Material handling costs were generated for the manually created load-plans and for the computer-assisted load-plans. The material handling costs and stability calculations generated by the container-terminal were used as a basis for comparison with the computer generated alternatives. (The results of this analysis are discussed below.) However, some containers that were re-handled (containers that arrived with the container-ship and were moved to new stowage positions before the vessel left the container-terminal) by the container-terminal were ignored in the analysis. The effect these ignored re-handles had on the comparison of computer-assisted and manually generated load-plans is unknown, but Martin *et al* claims they are not significant.

5.3.2.4.2 Stability results

In the evaluation of the computer-assisted results, the two measures used to check the stability of the container-ship were its metacentric height (GM) and trim. Ranges of acceptable values for the GM and trim of the container-ship were supplied by the vessel's operator. Stability calculations for the computer-assisted stowage-plans were not always reported by Martin *et al* as being within the desired range. The stability calculations of the computer-assisted stowage-plans were within a few percent of the manually generated stowage-plans. The GM values for the computer generated stowage plan were between 3.7% to 1.1% under those generated for the manually created plan. Trim values ranged from 1.9% over to 7.1% under those generated for the manual plan. Container terminal planners were reported as finding this difference acceptable.

5.3.2.4.3 Material handling results

The influence that a load plan can have on material handling time is attributed by Martin *et al* to five discrete factors:

- the number of transtainer moves;
- the distance a transtainer travels within yard sections;

- the number of times a transtainer has to move between adjacent yard sections;
- the number of times a transtainer has to move between non-adjacent yard sections; and
- the number of times a container is re-handled.

Time studies of the above factors generated average time estimates, reported by Martin *et al*, for the handling factors of a single container (see Table 5-4).

Process	Time estimate
Transtainer fixed time/move	19.86 seconds
Transtainer movement time	1.56 seconds per row
Adjacent section change time	15.09 seconds per change
Non-adjacent section change time	162.94 seconds per change
Re-handles	157.92 seconds per re-handle

Table 5-4 Transtainer time estimates

The material handling costs for the above five factors in the computer-assisted and manually-generated load-plans are compared in Figure 5.3.

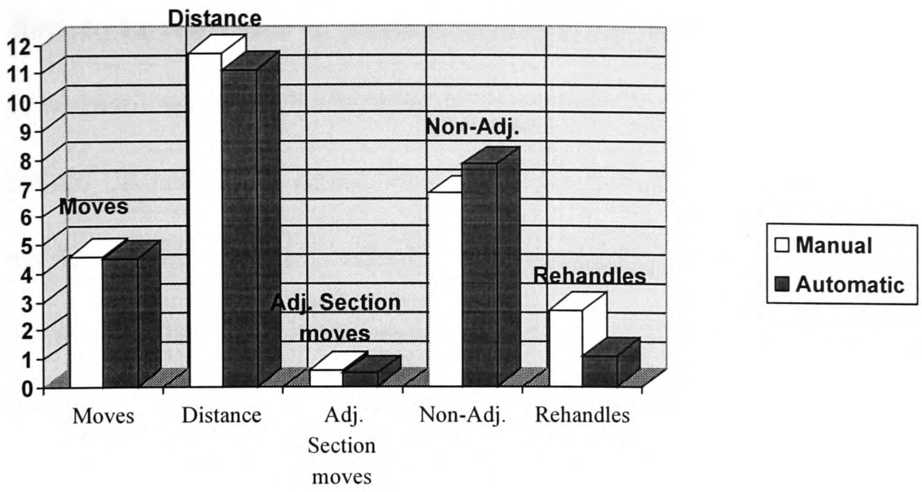


Figure 5-3 Material Handling results

An examination of the chart in Figure 5.3 reveals that the time attributed to the first four factors using the manual method (moves, distance, adjacent section changes and non-adjacent section changes) were comparable this those using the computer generated method. However, the computer generated results for factor 5 (time spent re-handling containers) was 41% lower than for the manually generated method.

5.3.2.4.4 Evaluation summary

Over the four cases analysed, the computer-assisted load-plans succeeded in reducing, on average, non-productive transtainer time by 4.8% and overall loading time by 0.6%. The computer-assisted plans can be prepared in up to half the time taken to produce plans manually.

5.3.3 Observations

The approach to automating the process of container-ship load planning due to Martin *et al* demonstrates how effective a heuristic can be in mimicking the decisions of a human planner. However, the container-ship planning process, which would actually have to be addressed by a human planner, was simplified greatly in order to allow this heuristic approach to be seen to be effective; Section 5.3.2.1 detailed this simplification. In particular, container-ship stowage planning involves dealing with exceptions to 'normal' cargo, which are ignored by Martin *et al.* Very few containers, in relation to the overall number handled during the loading of a ship, are of a hazardous nature. However, exactly how these containers are dealt with is of concern to the shipping operator. Very often hazardous containers are allocated specific stowage locations by the shipping operator. Account must be taken of these specific allocations during the load-planning process, because the exact position and time of loading will affect transtainer and crane utilisation. Similarly, out-of-gauge containers tend to be few in number, but they have a disproportionate affect on stowage planning and will very often be allocated specific stowage locations; these allocations will also have to be included in the load-plan.

Martin *et al* base the success of the method upon the assumption that it would be used for load-planning of newer, dedicated container-carriers with sophisticated ballast tanks which permit stability to be adjusted after loading. This in itself virtually invalidates the method since a large number of older container-ships could not be loaded. In addition, this can be viewed as an abuse of modern technology, as the ship-operator will not want to carry any more ballast than is necessary. It is likely that the result of reducing rehandles has been achieved partially by the poor use of ballast. The result that the computer model generates fewer re-handles can not necessarily be taken as evidence of having generated a better load-plan.

Finally, Martin *et al* makes no attempt to integrate allocation of containers to yard locations or to deal with containers that arrive late; all containers are assumed to have arrived and been allocated yard slots prior to planning. The positioning of containers within the yard has a direct bearing upon the efficiency of their heuristic and must be regarded as an important factor in the automation of the planning process.

5.4 Mathematical modelling

This class of automated planning processes includes work carried out exploring the application of linear programming to container-ship stowage. [36] Those practising this method of solving the stowage problem have over trivialised the constraints. These models have incorporated too many simplification hypotheses, which have made them unsuitable for practical applications.

5.4.1 Stowage container planning: a model for obtaining an optimal solution

Here, the multiple port stowage planning problem is approached using integer programming - a method for dealing with combinatorial problems that has close links with linear programming where integer variables, taking the values 0 or 1, are introduced in order to produce an integer programming formulation. [51] In the case of container-ship loading/unloading, an assignment of 1 means the slot is occupied and 0 means that it is empty. Botter [36] provides a mathematical model for describing the entire stowage problem, over multiple ports. This model includes information, at each stage of a ship's journey, on container destination, in which locations containers may be placed, and on circumstances under which unloading of containers will cause re-handling of other containers.

Botter specifies a number of shipping operator requirements that were taken into consideration when finding a solution using the specified mathematical model, namely:

- that no ship stability and stress constraints be violated;
- that container restows be minimised;
- that cargo is stowed so that ventilation, refrigeration and dangerous cargo requirements are obeyed;
- that ballast be kept to an absolute minimum;
- that longitudinal crane movement be minimised (and, by implication, hatch cover removal is kept to a minimum).

Except for some constraints, related to ship safety (and, presumably, problems relating to container size), Botter reports that the model can, in theory, be used to find an optimal solution for the stowage problem. However, Botter also admits that the combinatorial size of the stowage problem, for commercial ship operators is too large to solve in a commercially viable length of processing time. (Effectively, the model includes all possible placements of containers, and therefore describes a search space which is too large for a solution to be found in a reasonable length of time.) Instead, the author proposes the use of the basic features of the theoretical model in two different approaches which can be used to find good, if not optimal, solutions to the container stowage problem in a reasonable length of processing time. These approaches employ heuristics (see Section 4.1.2) to reduce the size of the search space of the theoretical mathematical model.

In this section, Botter's theoretical mathematical programming model is outlined, and a brief description of the (theoretical) means of solving the problem is offered. A brief discussion of the mathematical model and related implementation issues then follows. Descriptions of how Botter proposes to simplify the theoretical problem, thereby avoiding implementation issues are then provided. Lastly, the effectiveness of the author's approach, and the usefulness of describing the problem of container-ship stowage using a mathematical programming model, are analysed.

5.4.2 Theoretical Mathematical Solution

This section describes the mathematical programming model due to Botter.

5.4.2.1 The mathematical model

The complete container loading and unloading process for several ports on the route is viewed by Botter as a succession of individual stages. Within the mathematical model all possible unloading and loading stages are considered; the model assumes that at each port, theoretically, all the containers aboard the ship may be unloaded (although re-handling all containers at a port would be a costly solution).

These variables are based on information concerning container destination, in which cells containers can be placed, and on circumstances under which the unloading of a container will cause re-handling of other containers.

5.4.2.2 Using the mathematical model to find a solution

Botter uses a function, which operates on the state of the model, to reflect the *cost* of each container movement. The function considers the number of restows along the route and the total longitudinal crane movement along the berth during the loading and unloading process. Minimising the value produced by this function provides an optimal solution to the container-loading problem. However, this minimisation must be achieved in a way that also allows a number of other constraints to be satisfied and in accordance with a number of assumptions. These constraints and assumptions are:

- that each container is allocated a single slot;
- that each container can be unloaded and reloaded at any, or all, ports between its origin and destination;
- that a cell can be filled or emptied once, at most, in each port;
- that it is assumed that only one crane is ever in operation (this in itself prevents the use of this model in the real-world), and so one cargo handling will occur at any time;
- that a container can be loaded only on other containers or the base of a stack;
- that the hatch cover must have been removed before a container can be stowed in a hold;
- that a container can only be unloaded if all containers above it and, where appropriate, the hatch-cover have been removed;
- that the weight of container stacks cannot break stack limits;

- that the transverse metacentric height constraint (see section 2.3.5.3) can not be broken;
- that it is assumed that no alteration will be made to the ballast tank condition during loading and unloading;
- that a maximum angle of heal limit is kept within;
- that trim constraints are kept within;
- that shearing force and bending moment constraints are kept within.

5.4.2.3 Conclusions drawn by Botter from the mathematical model

The combinatorial size of the stowage problem described above is dependant upon the number of cells (container stowage locations) within the container-ship. The mathematical model of the stowage problem for a container-ship with only 1000 of these stowage locations calling on 4 ports would require nearly 10^9 variables and approximately 10^6 constraints, assuming the ship sails fully loaded. [36] This represents an extremely large state-space (see Section 4.1.1) to search. Therefore, it is not feasible to expect to reach a global optimum solution to the stowage problem in a commercially viable length of time. The mathematical model can not implemented in any practical way.

Given the combinatorial complexity of the problem, Botter developed two methods, using the above mathematical model as a basis, for reaching a solution that is not necessarily the optimum. These are described in the following section.

5.4.3 Non-optimal solutions to the stowage problem

Botter takes two distinct approaches to implementing parts of the theoretical mathematical study of the problem outlined above: Decomposition, and the Implicit Enumeration Algorithm.

5.4.3.1 Decomposition

Given that the complete mathematical model is too complex to solve, Botter decomposes the stowage problem into two, smaller, sub-problems, namely:

- an assignment problem;
- a sequencing problem.

The complete model is simplified further by classifying containers.

5.4.3.1.1 The assignment sub-problem

The solution of this sub-problem produces a plan of the container-ship at the end of the unloading and loading phases at each port on the route. For a loading phase, the solution generated shows the cell to which each container is allocated, without keeping track of the loading sequence (unlike the case of the complete model). Also, this method shows which containers are removed from their respective cells, again without showing the unloading sequence. Solving this sub-problem requires minimising overstows (while ensuring that the constraints specified earlier are still satisfied), thereby reducing the numbers of variables and constraints which must be considered.

5.4.3.1.2 The sequencing sub-problem

Here, the objective is to determine the optimal loading or unloading sequence between two stowage states. It is known *a priori* (from the solution of the assignment sub-problem) that a cell will contain a given container and the object is to determine at what stage during the loading sequence the container will be placed in that cell. Therefore, the solution of this sub-problem requires only minimisation of the longitudinal crane movement, again reducing the numbers of variables and constraints that must be considered.

5.4.3.1.3 Container classification

A combination of the solutions to the assignment sub-problem and the sequencing sub-problem produce a solution to the stowage problem defined by Botter. The production of this solution can be made simpler by container classification. Container classification is applied to the mathematical model. Rather than considering which cells each individual container could be placed in, it is assumed that a cell could be filled with any one of a class of containers which share the same type, origin, destination and weight range values. This greatly reduces the number of variables in the mathematical model.

5.4.3.2 Implicit enumeration algorithm

The general implicit enumeration algorithm is an alternative method for reducing processing time for combinatorially large problems. The complete mathematical model includes all possible solutions (successful and unsuccessful) effectively describing a state-space for the problem (see section 4.1.1). Implicit enumeration effectively prunes part of the state-space (see section 4.2.3) by considering some paths through the state-space to be unpromising. Partial solutions are abandoned (considered no further) if one of the two specified criteria are met. One criterion is that a partial solution violates a constraint. The second criterion is that there is another partial solution in the state-space, which advances the solution to the stowage problem equally far but at a lesser 'cost'.

5.4.3.3 Use of heuristics within the model

A number of heuristics (see Section 4.1.2) for loading container-ships were incorporated into the model. How these heuristics were used within the implicit enumeration algorithm is not made clear, but Botter reports that they concern:

- assigning many containers sharing the same destination to the same bay;
- if containers with different destinations are assigned to the same bay, then the containers with the farther destination port should be placed under those with the nearer destination port;
- during loading, the sequence of bays to be loaded is chosen so that trim constraints are not violated.

The use of these heuristics would reduce the size of the state-space.

5.4.3.4 Reported results

The two methods described above, decomposition with container classification (section 5.4.3.1.3) and implicit enumeration (section 5.4.3.2) were applied to a 740 TEU container-ship calling on four ports with a global supply of 1400 containers, using the linear programming language TEMPO and a mainframe computer. With the first method, only very small sub-sets of the overall problem could be solved. More success was found with the second method which Botter reports, provided a solution to the entire problem. [36]

5.4.4 Observations

The ideas presented by Botter are worth considerable discussion. Although the main shipping operator requirements have been clearly identified by Botter, there are still some inherent weaknesses in the method which are highlighted in this section. In addition, a number of conclusions not mentioned by Botter, can be drawn from the paper; these are discussed below.

5.4.4.1 Problem scope

Although Botter demonstrates that the size of the general container stowage problem is large, the methods of solution presented in the paper [36] rely on many simplifications to the problem. Firstly, the range of cargo types (described in Section 2.1.3) included in Botter's model has been kept deliberately narrow. In particular, cargo that requires ventilation, refrigeration or special segregation is ignored; *special* containers, such as hazardous types, are removed from the model because to include them would mean developing the mathematical representation of the problem so that it would become even more computationally difficult to solve.

When hazardous cargo is present, it becomes necessary to consider relative positions of all hazardous cargo (due to rules of segregation; described in Section 2.1.3.3). Failing to include hazardous cargo information in the model would introduce a requirement for all stowage solutions generated to undergo rigorous manual scrutiny and adjustment.

Secondly, the variety of dimensions of containers (described in Section 2.1.2.3) considered by the system has been reduced. That only containers of either 40' or 20' length can be processed by the algorithm is a deliberate design consideration which prevents the handling of a wide range of over length containers being processed. However, the heights and widths of containers that can be accepted by cells in Botter's model are not mentioned. It seems reasonable to assume, therefore, that each container in Botter's model has standard dimensions of 8'6" in height and 8' in width. This simplification of container dimensions excludes a wide range of cargo types, classified either as 'out-of-gauge' or of a different standard size (such as 10' and 30' lengths and 8' and 9'6" heights), which would frequently occur in real-world container stowage. It is clear that the type and dimensions of container has been reduced so that problems associated with encoding this approach could be avoided and the computational complexity be reduced. This simplification also precludes the user of Botter's system for any real-world application.

Lastly, the shipping operator objectives identified by Botter (described in Section 5.4.1) are insufficient and are not adhered to. While Botter's model does ensure that ship stability and stress constraints are met and container restows are minimised, other shipping operator objectives are not properly addressed, namely:

- special consideration for ballast, although recognised as being important to the shipping operator, does not form part of the mathematical model;
- although minimising longitudinal crane movement is of importance and is included in the global objective function its importance is incorrectly attributed purely to sequencing of containers when it should be included when assigning containers as well.
- Longitudinal crane movements may have inherent hatch-cover manipulation associated with them. Therefore, not considering crane movement when simply minimising restows may result in a number of hatch-covers being handled cost ineffectively.

In addition, since it is made clear in the model that only one container is handled at any time, Botter has assumed that only one crane is in operation at any port (illustrated in Figure 5-4). This is unrealistic, and has consequences for the load sequences produced and assessment of cost.

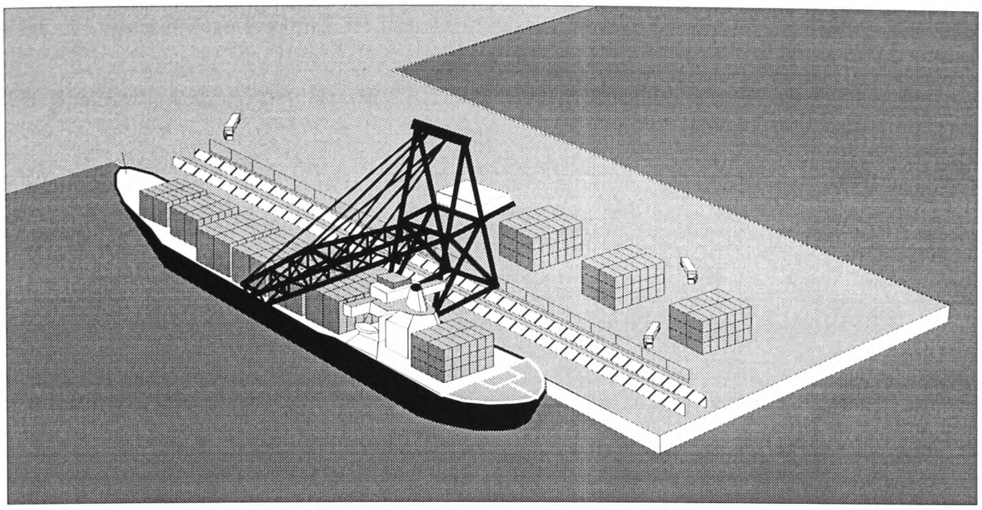


Figure 5-4 A single crane berth at a container-terminal

5.4.4.2 Implementation of the Decomposition approach

In this section, the decomposition approach to simplifying the complete mathematical model (explained in Section 5.4.2.1) is discussed. Decomposing the problem into assignment and sequencing sub-problems allows the conflict between the container-terminal planners and shipping operator planners to be highlighted and emphasises the inherent weaknesses and strengths in Botter's model.

5.4.4.2.1 Assignment of containers to cells

Botter has greatly reduced the scale of the assignment problem by only addressing the assignment of two basic types of container. Botter goes further in reducing the computational complexity of the assignment problem by removing the need for generating a specific sequence for container handling.

The specificity of container to slot assignment by the shipping operator varies considerably. In practice, it appears to be advantageous to reduce the specificity of container to slot pre-planning to an absolute minimum in order that the container-terminal planners can enjoy as much room for manoeuvre as possible when loading the container-ship.

Two reasons for this flexibility being granted by the shipping operator are:

- very often specific containers will not have arrived at the container-terminal until after loading has commenced;
- the terminal-planner will be able to arrange the loading of a vessel so that the terminal's material handling costs are minimised (reduced costs are passed on to the shipping operator in a very competitive market).

Therefore, not including container sequencing within the global model is advantageous to the container-terminal planners. However, not including sequencing in the Botter model reduces the solution of the model to that of simply minimising container restows.

Whereas, not generating specific sequences of container movements is advantageous to the container-terminal planner, simply reducing the objective of stowage planning to that of reducing container restows will ultimately lead to very poor block stowage and a gradual degradation, or fragmentation, of container placements by destination. Fragmentation of container placements within container-ships will result in excessive longitudinal crane movement and opening and closing of hatch-covers. Botter attempts to address this problem by the use of heuristics, although how this is done is not made clear. A fuller discussion of how heuristics have been implemented follows later.

5.4.4.2.2 Sequencing of container movements

The method chosen for generating a container handling sequence is inherently weak as it does not take into account a number of factors important to the container-terminal planners. Whereas reducing longitudinal crane movement is of importance, the model restricts this to one of reducing the movement of a single crane and no provision is made for container-terminals where two, or more, cranes are used to process containers. Containers in the yard will no doubt be placed in locations that, should the sequencing algorithm suggested by Botter be used, will result in the container handling equipment (usually a combination of transtainers and trucks) travelling a far greater distance and, as a consequence, take more time leading to a longer time spent at the container berth. Spending as little time as possible docked is of prime interest to the shipping operator as docking fees will have to paid accordingly. Therefore, since container sequencing is an important consideration, Botter partially addresses this by treating containers in a more abstract way, by grouping them into classes, introduced in Section 5.4.3.1.3 and explained in the following section.

5.4.4.2.3 Container grouping by class

In an attempt to overcome some of the observations made above, an important step is taken, that of grouping (classifying containers by general type, destination and weight and associating a whole class with a possible stowage location). Whereas the inherent practical weakness of not accommodating multiple cranes is not considered, the basic principle of allowing as great a flexibility to the container-terminal planners as possible is addressed. However, in chapter 6 it will be argued that it is necessary to take a further step of abstracting the container-stowage space on the ship; specifically that it should be viewed as areas to be filled. The problem solutions produced by Botter's decomposition approach, reduced to that of placing the most common amongst the myriad of container types, will never have any real world application.

5.4.4.3 Implementation of the Implicit enumeration approach

In this section, the implicit enumeration approach to simplifying the complete mathematical model (explained in Section 5.4.3.2) is discussed. The so called implicit enumeration algorithm described actually appears to be a widely used branch and bound search (introduced to the reader in Section 4.2.3) where the direction of the search is controlled by the introduction of heuristics. Botter does not describe the exact method in full. The three components of the algorithm briefly covered in the paper are discussed in the following sections.

5.4.4.3.1 How constraint handling is used to reduce the state-space

The first criterion described by Botter when pruning the current state-space of the problem is to remove partial solutions that violate a constraint without exploring further container placements. This approach is not unusual and is often adopted by practitioners attempting to solve combinatorially large problems. Two general types of constraints will be encountered when developing a solution, those that prohibit further development and those that do not. Encountering constraints, such as a container stack weight limit being broken, will always result in an invalid solution where there is no reason to explore additional stages further. However, failing constraints, such as trim and heeling restrictions, would not prevent further exploration as new container placements may counteract the earlier problems and lead to acceptable solutions. Botter fails to describe whether both these types of constraints are treated the same way in his paper.

Whereas not developing partial solutions that fail a constraint will result in a much-reduced problem, further expansion may produce a valid solution at a later stage in the search process. Therefore, the probability of finding anything close to the optimal solution is also reduced. This is referred to as the horizon effect (explained in Section 4.2.2). Many practitioners attempt to overcome the horizon effect by continuing to explore partial solutions a given number of stages further over the 'horizon' in an attempt to escape local problems and ultimately find a globally optimum solution.

5.4.4.3.2 How metrics are used to prune the state-space travelled

The second approach adopted by Botter for pruning the state-space of unpromising solutions involves comparing the value generated by the objective function for a given branch with all other branches at the same level in the tree and removing it if another solution can be found that 'costs' less to arrive at. This is another common approach for reducing the state-space of a problem. How effective this heuristic is depends greatly upon the problem and objective function used. The result generated by the objective function used in the model under consideration is material handling cost. The cost and relative worth of each partial solution will vary greatly as it is developed. Therefore, discarding partial solutions that, at a given stage, cost more than the best found up to that stage will also suffer from the horizon effect. The trade-off between processing time and reaching a solution closer to the optimum is an important design consideration when considering combinatorial problems.

5.4.4.3.3 Use of heuristics

Botter alludes to other heuristics, but exactly how these are incorporated into the mathematical model is not made clear. The heuristics mentioned are sensible and model the actions of a human planner. However, the heuristics also highlight the limited scope of the problem tackled, particularly with respect to the assumption of a single crane being in use.

5.4.5 Conclusion

The work of Botter [36] clearly demonstrates the difficulties associated with attempting to assign specific containers to stowage locations on a multi-port trade route. Botter fails to develop a workable system for the solution to the general container stowage problem because of the short-cuts and over-simplifications incorporated into the model. However, the work is of interest for two main reasons. Firstly, Botter identifies and proves the computational complexity of the continuous port stowage problem. Secondly, Botter provides a strong development along the path to success in classifying containers into general groups, rather than attempting to associate specific individual containers with slots. This abstraction of the problem is of particular interest as it models the human planners approach to tackling the problem.

In Chapter 6, the idea of modelling the human planners approach will be extended. In addition, it will be argued that Botter's approach demonstrates that the combinatorial difficulties are partly a result of the cargo space abstraction chosen at the design stage and the inadequate conceptualisation of the global stowage planning task. A greater degree of abstraction of the global stowage problem is required in order that the combinatorial problems can be surmounted.

5.5 Decision support systems

This group is entirely separate to the first three categories described above. No effort is made here to automate the generation of stowage solutions. Instead, sets of tools are made available to the users that assist in the generation of stowage solutions. [1,43,44,45] The partnership of tools and human expertise has given the best commercial results to date. An example of a decision support system is outlined below.

5.5.1 Decision Support for Container-ship Stowage Planning

A typical decision support system automates data management functions and provides computational capabilities that allow the stowage planner to rapidly assess the impact of stowage configurations on vessel intact stability and stress parameters. The following section highlights the work done by Saginaw in developing a stowage planning computerised tool that assists the human planner.

5.5.1.1 Objectives

Three objectives that support the general goal of providing the planner with a decision support system are introduced by Saginaw, namely:

- that the developed system should fit into a generic microcomputer since few shipping operators possess larger, more powerful, computer systems;
- that the computerised stowage tool should be able to accommodate a wide variety of different types of container-ships;
- and, that the computerised system should not disrupt the planner's processes but, rather, integrate naturally into them.

5.5.2 General description

The stowage-planning tool developed by Saginaw is microcomputer-based and uses interactive visual graphics to allow the user to experiment with a variety of stowage configurations. The system is able to display, all, or part of, the container-ship's cargo space so that the planner can more easily conceptualise and alter the stowage configuration of cargo. After each alteration of the cargo stowage configuration, the computer-system automatically updates the ship's intact stability and stress parameters (shown in Table 5-5).

Displacement	Actual metacentric height	Actual shear force
Angle of heel	Required metacentric height	Maximum shear force
Draft forward	Metacentric height margin	Shear force margin
Draft aft	Actual bending moment	
Trim	Maximum bending moment	
	Bending moment margin	

Table 5-5 Vessel intact stability and stress parameters

The planner is able to adjust the tankage configuration of the ship. Tanks are divided into three types:

- fuel oil;
- ballast;
- and other.

Naturally, the intact stability and stress parameters are updated accordingly after each alteration to the tank configuration.

5.5.3 Conclusion

The introduction of a computerised planning tool greatly facilitates the planning process permitting more flexibility within the planning process to experiment with a variety of different configurations. The planner's scope for experimentation while using a paper-based system is, naturally, curtailed. Therefore, the introduction of computerised tools, although still in its infancy with approximately 80% of shipping operators still using a paper-based system [12], has resulted in reduced handling costs and more efficient cargo transport.

5.6 Rule-based expert systems.

In response to a perceived inability of conventional optimisation methods to provide optimal cargo stowage due to the considerable number of possible loading strategies, some researchers have turned to exploring the effectiveness of rule-based expert systems.[1,45,46,52, 53] The objective of most of the researchers [1,45,52,53] exploring rule-based expert systems is to produce a system that provides decision support for a qualified stowage-planner. However, one example of this type of approach where the whole planning task is automated and does not rely upon the interaction of a user is the work attributed to Sato *et al.* [46] Although the work completed by Sato *et al* involves planning the cargo stowage for oil-tankers it is significant because it demonstrates a production-rule (see Section 4.3) based approach and highlights the advantages of a more generalised planning strategy.

5.6.1 Expert System for Oil Tanker Loading/Unloading Operation Planning

The application of rule-based expert systems in producing solutions to loading problems is well illustrated by the approach due to Sato *et al.* [46] In particular it illustrates how container-ship cargo space can be generalised and modelled without affecting intact stability calculations.

5.6.1.1 Introduction

Sato *et al* [46] describe a prototype rule-based expert system which solves a problem which is in some way analogous to the general container-ship loading problem. Sato *et al* provide a rule-based solution that:

- solves the oil-tanker cargo arrangement problem where different grade oil is assigned to different tanks;
- acts as a repository for planner's experience and knowledge.

5.6.1.2 System Objectives

The main objectives, set out by Sato *et al*, are:

- that any draft limitation (explained in Section 3.3) at the loading port be met;
- that a proper condition (taking into account trim, propeller immersion *etc.* explained in Section 2.3.5) for a laden voyage be generated;
- that longitudinal strength limitations (such as bending moments and shearing force, explained in Section 2.3.6) be met;
- that cargo left-behind be minimised;
- that the number of empty cargo-tanks be maximised.

5.6.1.3 The oil-tanker loading process

Sato *et al* have identified a two part process that models the load planning undertaken by an oil-tanker planner. The first part of the planning process undertaken sees the planner conceptualising a variable number (dependant upon the planner) of different stowage configurations each of which is ranked according to general efficiency. After selecting the most promising looking stowage configuration, the planner then manipulates this general plan by moving cargo around the ship and adjusting ballast until all constraints are satisfied.

5.6.2 Description of the oil-tanker planning system

The planning operation identified by Sato *et al* has been implemented as two distinct processes combining undirected (described in Section 4.2) and directed (described in Section 4.3) search techniques. Both of these processes are described in full in the following sections.

5.6.2.1 Undirected search and oil-tanker load planning

An exhaustive search is made of all possible loading patterns, each of which is ranked according to cargo left behind. The example general arrangement (see Section 3.3.2.1) given by Sato *et al* has 11 tanks allocated for the transportation of oil. Sato *et al* generates loading patterns where two different types of oil are to be transported. Several of the best-ranked patterns generated above are compared according to maximisation of empty tanks. This two part ranking results in a set of loading arrangements ranked according to how little cargo is left behind and how few of the ship's tanks are used. The intact stability for the highest ranked cargo arrangement is calculated and used to direct any required rearrangement of the cargo. Should the best generated cargo arrangement fail any intact stability and stress calculations then planning enters a second phase where a set of rules are used to modify the arrangement (described in detail in the following section).

5.6.2.2 Directed search applied to the oil-tanker loading problem

Having selected the best solution based upon the criteria of maximum cargo carried and minimum number of tanks used, intact stability is then calculated and used to direct the distribution of ballast and, perhaps, some redistribution of cargo. An expression of the ship's state is derived (shown in Table 5-6) based upon the result of the intact stability calculations. This state value is used to help determine the best modification of the ship's state.

Ship's State	Value
Extremely over the upper limit	+3
Over the upper limit	+2
Just below the upper limit	+1
Within limits, no problem	0
Just below the upper limit	-1
Below the lower limit	-2
Extremely below the lower limit	-3

Table 5-6 Expression of the ship's state

The example given by Sato *et al* has fifty load modifying operations (alterations to ballast or relocation of cargo) associated with it. Each of these operations will affect up to twelve state items (given in Table 5-7) that make up the intact stability of the ship.

Operation/ State items	Shift from 5OT(C) to 3OT(P/S)	Shift from 2BT(P/S) to ABT	Ballasting into 4BT(P/S)
Trim	-	++	0
draft aft	-	+	+
draft fore	+	-	+
draft mid	0	0	+
Bending moment	+	-	+
Shearing force at 68	0	-	0
Shearing force at 71	0	0	0

Operation/ State items	Shift from 5OT(C) to 3OT(P/S)	Shift from 2BT(P/S) to ABT	Ballasting into 4BT(P/S)
Shearing force at 79	+	0	-
Shearing force at 86	+	0	+
Shearing force at 95	-	0	0
Shearing force at 102	0	++	0
Shearing force at 110	0	0	0

Table 5-7 Example operation/state-item vectors

The operations given in Table 5-7 refer to either a movement of existing oil or ballast or the addition of ballast, *e.g.* Shift from 5OT(C) to 3OT(P/S) means move oil from number 5 central oil-tank to number 3, port or starboard oil-tank, each of which can be found in Figure 5-5. The number next to the shearing force state items (*e.g.* 68, 71 *etc.*) refer to positions on the ship, which can also be seen in Figure 5-5.

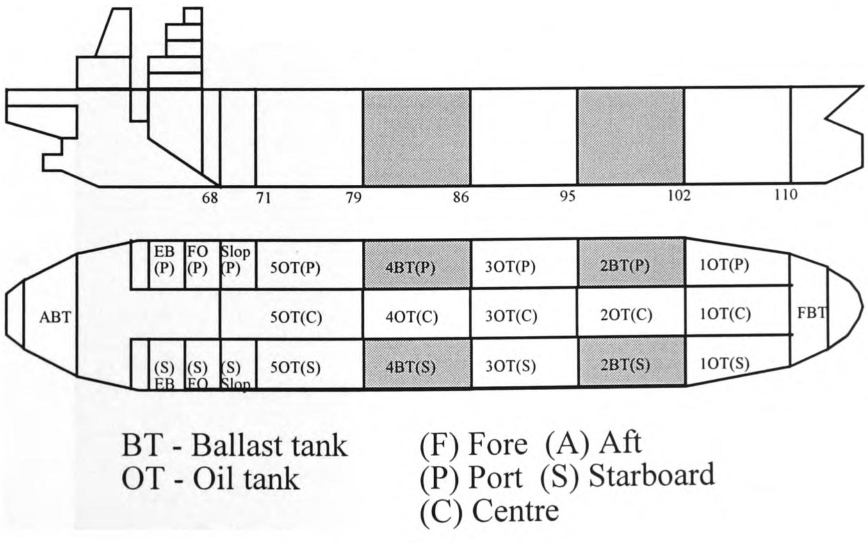


Figure 5-5 Sample General Arrangement

Each operation, when cross-referenced with a state item, has a symbol (explained in Table 5-8) representing its affect upon the intact stability of the ship.

Symbol	Degree of effect
++	Increases the state value extremely.
+	Increases the state value.
0	Has no major effect to the state value.
-	Decreases the state value.
--	Decreases the state value extremely.

Table 5-8 Expression of effects of operations

The symbol found by cross-referencing the operation with the state-item is compared to the ship's state value indicating its relative desirability (illustrated in Table 5-9). The effect of the operation upon each of the state-items is determined and used as a basis of comparison for determining which operation to perform.

Ship's state value	Effect of operation	Score
3	--	+5
3	-	+3
2	-	+2
1	++	-3
1	+	-1
0	Whatever	+
-2	+	+2
-2	-	-10
-2	++	+3

Table 5-9 Evaluation scores

Each of the operations is instantiated within the knowledge-base of the expert-system in the form of rules. In addition to rules for each operation, additional rules that constrain what operations can be performed are also instantiated. One of these rules that prevent impossible operations from occurring takes the following form:

IF the operation being considered is shift from tank A to tank B

AND

tank A is empty OR tank B is full,

THEN the operation is impossible

Sato *et al* use a forward chaining (described in Section 4.3.3.2) inferencing strategy to optimise the storage configuration of the ship. Exactly how inferencing takes place is not made clear but probably takes the following form:

- all possible moves (those that are not constrained by the system, such as the one given in the example above) and their associated scores are determined;
- the conflict resolution set (described in Section 4.3.3.2, in this case the set of allowable moves where the conflict arises from deciding which move to make) is ordered according to each score;
- the best move from the conflict resolution set is performed;
- if the new stowage configuration is still unacceptable then this process is repeated, otherwise;
- The final solution is presented to the user.

5.6.3 Conclusion

Although tanker cargo allocation planning does not present the same degree of complexity as the container-ship stowage problem a number useful conclusions can be drawn from this work. Combining undirected and directed search is novel and appears to model the thought processes of a planner very well. Since generating all permutations for the relatively limited state-space of the oil-tanker loading problem and thus finding a optimum solution does not appear to be a difficult proposition, the importance placed upon rule instantiation and separation of inferencing from the knowledge-base clearly indicates that accountability to the user is of prime importance to the writers. The issue of accountability to the end-user often dictates which method of implementation is selected by the knowledge engineer. The system developed by Sato *et al* generates a clear audit trail of how and why a solution was generated and has the advantage of being able to act as a repository for accumulated expertise. This repository of expertise safeguards the employers position should the human expert move on and the system can act as a training tool for stowage-planners.

The optimisation process attempts to find a solution with acceptable intact stability by altering the most promising stowage pattern taken from the initial exhaustive search of all possible configurations. Should no improving move be found, the optimisation process is restarted with a new solution taken from the pool of most desirable configurations generated during the exhaustive search. Therefore, the optimisation process is, in essence, a hill-climbing search where the climb is restarted from a different point each time it fails to reach a global solution. However, restarting the whole optimisation phase with another solution generated during the initial creation of all possible stowage patterns may not be required. Instead, it may be desirable to continue the search by making further moves in the hope that, after an initial descent, a new hill can be found and climbed to a satisfactory conclusion. As the model stands, no attempt is made to minimise the amount of ballast carried. Minimising ballast is not seen as an important requirement with the objective being to find a solution that minimises the number of oil-tanks used whilst maximising the amount of cargo carried. This will inevitably mean that some solutions generated will require the costly and wasteful transport of unnecessary ballast. Including the placement of ballast into the initial phase of exhaustive search would have the disadvantage of increasing the final number of solutions generated but would produce a final solution, that is closer to the optimum, where left behind cargo is minimised, oil-tank usage is minimised and the amount of ballast is minimised. Sato *et al* appears to have rejected this approach because of combinatorial and computational complexity. The prototype system developed by Sato *et al* has the advantage of being accountable to the user but has the disadvantages of not minimising ballast and having no guarantee that an optimum solution can be found. The approach is of special interest since overall computational complexity is reduced by decomposing the planning process to a first phase generalisation where intact stability and ballast are ignored and a second phase optimisation process where the initial generalisation is progressively improved by the selection of moves. The importance of these general principles, when considering the container stowage problem will be explored in Chapter 6.

5.7 Summary

A variety of computerised cargo-ship-planning applications and tools have been considered in this chapter. When considering the applications related to container-ship stowage, important common denominators can be identified. These similarities highlight the inherent difficulties associated with each approach to solving the container-ship stowage-planning problem.

Each approach has:

- highlighted the combinatorial complexity involved with considering each conceivable stowage configuration;
- identified domain features and constraints;
- simplified the problem by only considering the placement of standard containers into standard cellular container-ship stowage locations (thus, disregarding out-of-gauge containers);
- simplified the problem by limiting consideration of special containers, such as making provision for power external power sources;
- simplified the problem by not making provision for the segregation of hazardous containers;
- not analysed how and why human planners prepare stowage plans, resulting in the above simplifications intended to reduce the computational and combinatorial size of the problem to a degree where the production of at least partial solutions are feasible.

General principles used successfully by the authors of the systems considered, and the lessons learnt from the difficulties raised above, were used in the development of the approach to the stowage problem in this project described in this thesis. The development of the approach taken here is described in the following chapter.

6 SOLVING THE DEEP-SEA CONTAINER-SHIP STOWAGE PROBLEM

6.1 Introduction

The literature search (the results of which are outlined in Chapter 5), provided the basis for key ideas for practical experimentation. This work was conducted in parallel with the work of devising *new* models for use in solving the deep-sea container-ship stowage planning. The studies and reasoning behind the eventual proposal of a model for solving the deep-sea container-ship stowage problem, along with the model itself, are presented in this and the following two chapters. This chapter presents a system overview that identifies the requirements for a solution to the stowage problem. Experimentation with different methods of modelling the relationships between physical structures (*that is,* the container-ship, containers and cargo) facilitated an in-depth understanding of the implications that underlie other authors' experiences. The theoretical model developed during the life-time of this project was, therefore, continuously revised as additional information became available.

6.2 System overview

The following section explains why the automation and optimisation of the stowage planning process is advantageous. General system objectives are then identified.

6.2.1 System advantages

The advantages that can be expected from an effective computerisation of the stowage planning process follow:

- A reduction of over-stowage (the placement of containers on top of containers destined for an earlier port, explained in Section 3.2.2) - thus reducing the number of re-handles. Re-handles are considered a major cause of wastage of time, resources and, ultimately, money. Better planning would reduce the number of container re-handles to an absolute minimum.
- Increased cargo-handling efficiency. The distribution of cargo between hatches would facilitate quick and efficient loading and unloading where the need for the removal of hatch-lids would be minimised. The intelligent separation of cargo between hatches would permit an efficient use of cranes.
- Increased vessel utilisation. It could be ensured that cargo would not be stowed in so poor a way that the loading of future cargo might not even be possible. An important issue here is the intelligent stowage of special cargo stowage.
- Increased managerial control due to a substantial increase in the amount of management information available for assisting strategic decision making.

- The retention of expertise that is currently hard to acquire. A very important product of computerising the process is the preservation of expertise that would otherwise be the sole domain of the human planner.

6.2.2 System analysis

The deep-sea container-ship cargo stowage problem has been described in Chapters 2 through 4. These chapters discussed the perspective of the planner. In this section, planning will be revisited from the perspective of the requirements of an automated system. The ways in which other authors have approached the automation of cargo stowage planning has been covered in Chapter 5.

By summarising the key issues addressed in chapters 2 to 5, it is possible to produce a list of the salient problem features that must be addressed by a computerised solution:

- a variety of different types of cargo (introduced in Section 2.1.3) are transported usually, but not always, in a container;
- the container is sometimes cargo specific (Section 2.1.2.4), and is increasingly likely to be of standard dimensions (Section 2.1.2.3);
- container-ships come in a variety of sizes (Section 2.2.1) and with a number of different internal and external configurations (introduced in Section 0);
- container-ship stowage planning is subject to strict rules concerning:
    - intact stability (described in Section 2.3.5) and stress (Section 2.3.6);
    - cargo segregation (Section 2.1.3);

  - restricted placement of cargo (in Section 3.4);
  - heuristics intended to reduce handling-costs (Section 3.3).
- container-terminals each have attributes that will vary (introduced in Section 2.4.1). These attributes include:
  - the number of cranes at each berth;
  - the financial cost to the shipping operator of each crane movement;
  - physical limitations upon the length of container-ship and height of on-deck containers that can be easily accommodated.

6.2.3 Stages of planning

Specific procedures for planning stowage of cargo on container-ships in each shipping company vary. However, a two-stage planning process is commonly employed referred to in this thesis using terms chosen by the author - strategic planning (introduced in Section 3.2.1) and tactical planning (Section 3.2.1). [43] An analysis of the 'tools' (the stowage planning documents described in Section 3.3.2) used by the human planner to perform the task of stowage planning is given below. This analysis reveals how the planning process is decomposed into sub-processes, each of which is more easily dealt with than would be the overall process.

6.2.3.1 Strategic planning

During the strategic planning phase, the human planner follows general guidelines for placing containers in specific areas of the cargo-space (described in detail in Sections 8.2.2.3 & 8.2.2.4). These guidelines divide the vessel into areas where containers of a specific class (defined by destination, container type and content) are grouped together. At this stage in the planning process, little detailed information about the cargo is available and the pre-planner depends upon forecasts taken from historical data. This generalised strategic planning process is divided into two sub-processes.

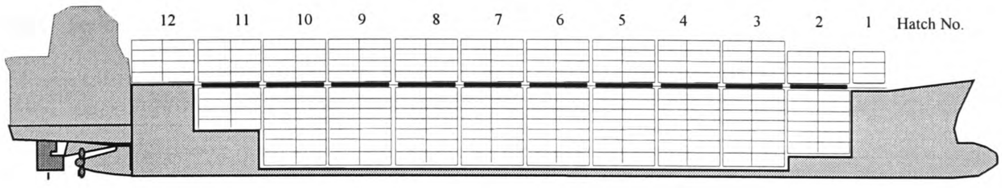


Figure 6-1 Sample General Arrangement showing hatch split

The first sub-process that facilitates the strategic planner's task involves the use of the document commonly called the General Arrangement (described in Section 3.3.2.1 and illustrated in Figure 6-1). The General Arrangement is used to allocate groups of containers, according to destination, length and content, along the *length* of the ship. By first considering the longitudinal stowage of containers, the planner's conceptual task is considerably reduced in scale.

The positions in which containers are placed within the General Arrangement is dependant upon the existence of other cargo for the same destination, the permitted length of container within each hatch, provision for special cargo (such as cargo that taints its environment) and the number of cranes that will be operating on the container-ship at the destination port (explained in detail in Sections 8.2.2.3 & 8.2.2.4).

The strategic planner's primary objective at this stage is to ensure that crane usage at the destination port is maximised and any constraints are met. Planning at this stage will result in containers generally being spread across the container-ship in as many hatches as there are cranes at the destination port. Sufficient space will be left between selected hatches to permit simultaneous operation of all cranes. The presence of existing cargo on the container-ship will often facilitate the planner's decision making process by indicating which hatch new cargo will be allocated to. Planning for placement of hazardous cargo that must be separated longitudinally is also assisted by the use of the General Arrangement.

The second sub-process performed by the planner involves making the general placements of containers described above more specific. In this sub-process, a second document is used - the Outline Plan (described in Section 3.3.2.2). The Outline Plan (an example of which is partially reproduced in Figure 6-2) is used to allocate containers, within the hatches indicated on the General Arrangement, to above or below deck stacks. A further consideration when using this document is the positions of hatch-lids. A *specific* container is not necessarily allocated to a cargo-space (in this case a group of above-deck or below-deck stacks that correspond to a hatch-lid), rather one of the containers of a specific general class will be allocated. The choice of which specific container, of the general class, is put in that location is made later in the tactical planning phase.

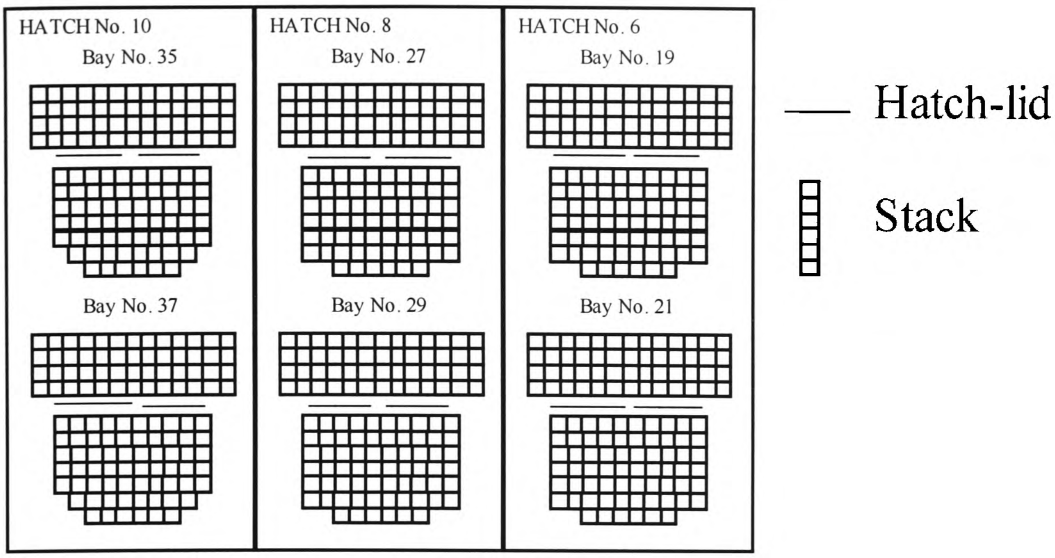


Figure 6-2 Partial reproduction of an outline plan

The primary objective when planning the stowage of containers on the outline plan is to minimise the removal of hatch-lids during the unloading process whilst minimising the amount of unused, below-deck cargo-space. Provision for the placement of containers with special requirements (such as requiring an external power source), the handling of out-of-gauge containers and the segregation of containers with hazardous contents can also be made. However placements of containers on the Outline Plan are left as general as possible.

Voyage number:______ Date:______ Port:_____/______
Discharging/Loading

210814	210614	210414	210214	210014	210114	210314	210514	210714
210812	210612	210412	210212	210012	210112	210312	210512	210712
210810	210610	210410	210210	210010	210110	210310	210510	210710
210808	210608	210408	210208	210008	210108	210308	210508	210708
210804	210606	210406	210206	210006	210106	210306	210506	210706
	210604	210404	210204	210004	210104	210304	210504	
		210402	210202	210002	210102	210302		

Bay No. 21
Under deck
8' 6"

Figure 6-3 Sample bay-plan

6.2.3.2 Tactical planning

In the tactical planning phase, the actual stowage locations for specific individual containers are determined, and a third document, the *Bay Plan* is used. During the tactical planning phase specific allocations of containers to stowage locations are recorded on individual Bay Plans (which are described in Section 3.3.2.3). An example of a Bay Plan is shown in Figure 6-3, in which the numbers (210814, 210812, *etc.*) represent stowage locations (slots).

The generalised stowage pre-plan of the Strategic Planning phase is used to direct these specific placements of containers to slots, as and when detailed information about each container becomes available. During tactical planning, a significant number of containers may still be enroute to the container-terminal; some operators accept containers as little as three hours before the vessel sails (significantly Japanese ports place a twenty-four limit [12]). For this reason individual Bay-Plans (described in Section 3.3.2.3) are often prepared incrementally with each completed plan being passed to stevedores, who attempt to load the container-ship as close to the plans as possible. Additional bay-plans are generated (or refined) as new containers become available for loading. Containers are generally placed within the cargo-spaces allotted for them during the strategic planning process. Care is taken to ensure that all constraints, such as hazardous segregation, upon container placements are adhered to. The specificity of the bay-plans generated by the shipping operators' planners will vary between operators. However, the container-terminal's planners will prepare detailed bay-plans that show the precise stowage configuration of the vessel, including details of any alterations to the shipping operators' instructions that may have been carried out.

The problem, then, is characterised by the relationship between containers, container-ships, container-terminals and an integrated planning process where the degree of flexibility given by the shipping operator planner to the container-terminal planner for making specific stowage decisions varies considerably.

6.2.4 Outline of the computerised system

The descriptions provided in chapters 2 through 5 constitute a knowledge elicitation exercise, the results of which were summarised in the previous section. What was learned from this knowledge elicitation exercise, and how it has influenced the design process, is discussed in the following sections.

6.2.4.1 System input and output

The envisaged system would require information about:

- the specific container-ship, including information about the cargo-space (such as maximum stack heights, weights and restrictions of container dimensions), information relevant to intact stress and stability, the location of power outlets for specialised containers and such considerations as location of crew quarters, engine-room and tanks;

- known cargo, including specifics about the type of cargo and the dimensions of the containers;
- how to segregate special types of cargo;
- expected (unknown) cargo, based upon statistical analysis of past voyages;
- the sequence container-terminals will be visited;
- the container-terminals to be visited, such as the financial cost of each container movement, number of cranes at each berth and special working arrangements of the personnel employed at the terminal.

These inputs are summarised in Figure 6-4. The system should be able to output a projected, generalised Outline Plan for each port-of-call on the voyage in addition to specific Bay Plans for the current port-of-call.

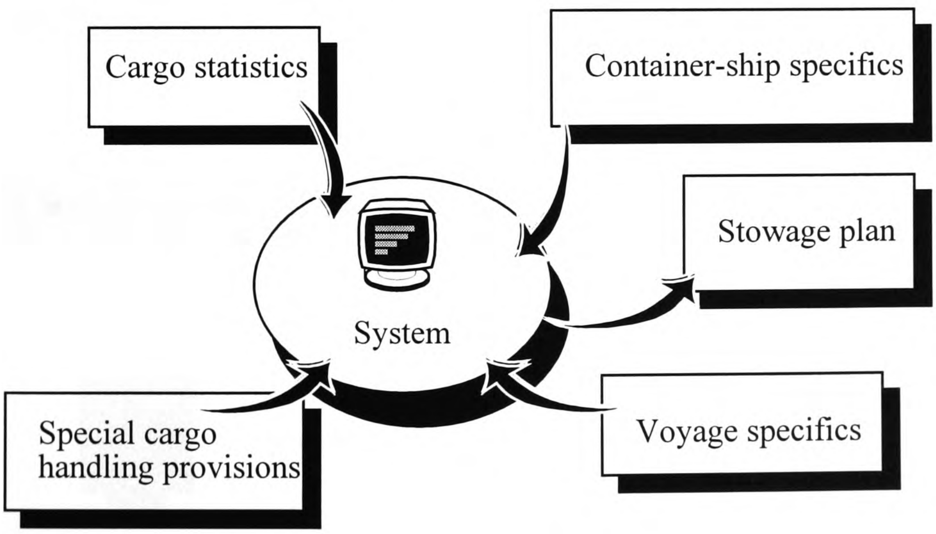


Figure 6-4 Computer system input and output

6.2.4.2 Processing requirements of the computerised planner

The proposed system should be able to perform both the strategic and tactical planning phases normally performed by a human planner. This is to say that the computerised planning system should generate stowage plans for a container-ship at its current port-of-call that optimally reflect expected transactions at subsequent port-of-calls.

The computerisation of the planning process should effectively model the thought processes undertaken by the human planner whilst undertaking the strategic and tactical planning tasks described in Chapter 3 and expanded upon in Section 6.2.2 (illustrated in Figure 6-5).

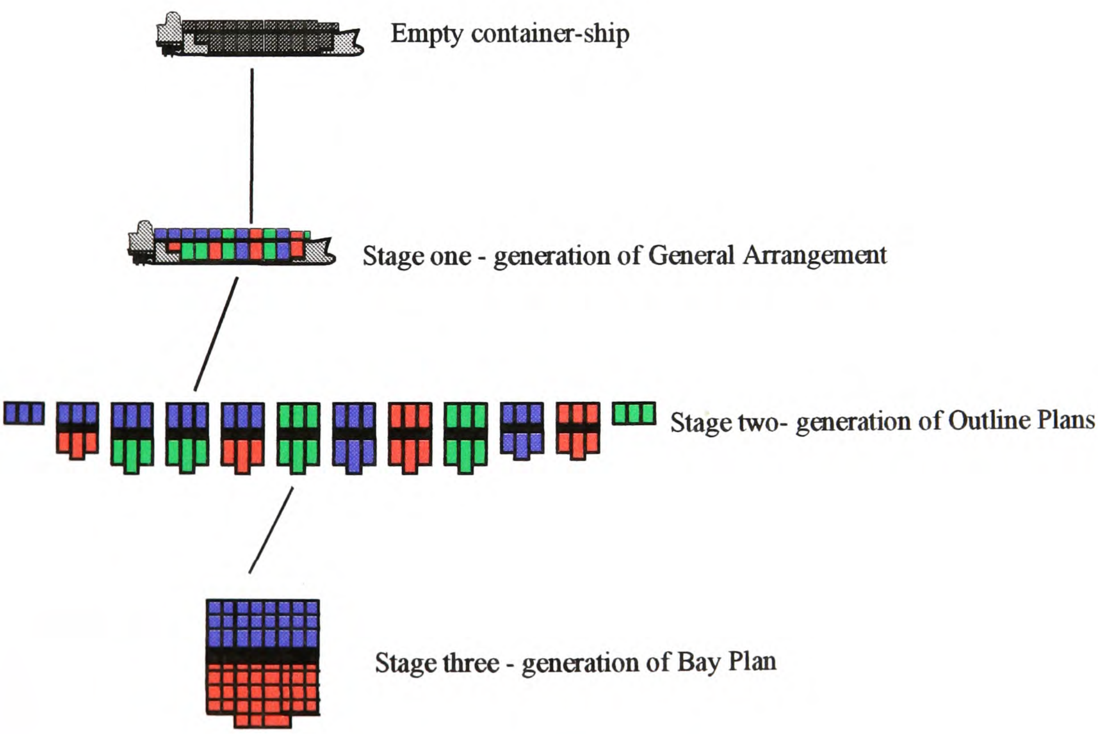


Figure 6-5 Three stage planning process

The output from the planning system represents the stowage pattern of a vessel leaving port. The computerised process should generate a *set* of stowage solutions that satisfy all given constraints and heuristics from which a best solution would be chosen. In theory, there will be numerous different solutions that satisfy all the given constraints. Each of these solutions has both an inherent short-term and long-term cost. The short-term cost is the expenditure of the stowage operation at the current port. Long-term costs are those which occur at subsequent ports as a direct result of decisions made at the current port of call. The future effects of current stowage decisions must be explored by simulating the planning operations at further ports. This process is iterative as the planning and simulation extends to the next, and further, ports-of-call. Since the whole process of planning will have to be repeated as the voyage is simulated data will be required periodically in order that the plan for the voyage can be updated effectively. Much of the information about cargo at future ports will be statistical and the quality of this information will determine the true effectiveness of the envisaged computerised system. Shipping companies already make extensive use of forecasts, so little difficulty is envisaged in finding appropriate statistical information. This information would reflect factors that influence port-turnaround-time. The system will have to be as dynamic as possible and it is therefore envisaged that extensive use of Electronic Data Interchange (EDI) will be made. Finally, the system will have to satisfy all the relevant constraints such as intact stability and hazardous cargo placement.

6.3 Conclusion

The stowage-planning task, as performed by human planners, exhibits two classic search elements: constraint handling and the use of heuristics. These heuristics and constraints should be treated as independent components within the planning system since these factors vary from operator to operator. The envisaged system is knowledge intensive, requiring a variety of different inputs. The system will require forecast information that will be provided by the shipping company. Therefore, some suitable interface and appropriate protocol will have to be developed in order that this information can be made available to the system. The system will require information about the containers that are to be loaded whether the information be actual or forecast.

The following chapter describes the development of the design for a computerised planning system that solves the deep-sea container-ship-planning problem. In particular, it demonstrates how general Artificial Intelligence problem solving algorithms can be employed for this purpose.

7 DESIGN PROCESS

7.1 Introduction

As explained in Chapter 6, the author undertook in parallel an on-going literature search and domain familiarisation. For this reason, revisions being made to models for solving the deep-sea containers stowage problem were incorporated as and when new insights into the problem were drawn. This type of incremental learning process lends itself well to a prototyping [74] approach to system design and implementation. The following sections describe how the proposed solution to computerising the stowage planning process evolved. In particular, the evolution of the data structures used to model the physical structure of a container ship is outlined. Careful reference is made to how the lessons learnt in parallel with the literature survey facilitated a deeper appreciation of important problems. These lessons lead to an approach to producing a system for solving the container-ship stowage problem explained in Chapter 8.

7.2 Initial conceptualisation of the problem

An initial examination of the deep-sea container stowage planning problem, based upon discussions with maritime personnel [6, 12, 68], resulted in research focusing upon cellular container-ships. This initial emphasis upon ships designed to transport containerised cargo was driven by a desire to benefit, from a design and implementation perspective, from an increasingly standardised inter-modal transportation system. A computerised solution to the deep-sea container-ship stowage problem was seen as requiring research into two distinct, but related, areas:

- the data structures used to model physical structures;
- and the procedural strategies used to produce stowage solutions.

There are a great variety of different types of deep-sea container-ship (introduced in Section 2.2.1) that ply the world's trade-routes. The production of a generic method for representing the important features (such as cargo-space geometry, tanks and crew quarters) of any container-ship, that could be used as a basis for exploring the cargo-stowage problem was seen as the first research objective. The following sections trace the evolution of the data structures that model container-ships experimented with by the author. These data-structures allowed application of problem solving algorithms, based around general Search theory introduced in Chapter 4, to the strategic and tactical planning phases described in Chapter 3, and expanded upon in Section 6.2.2. The procedural strategies used to produce stowage solutions will be dealt with in detail in Chapter 8.

7.3 Exploitation of cellular cargo-space

As a first stage in producing data structures suitable for solving the container-ship stowage problem, it was necessary to consider the relationship between standardised containerised cargo and purpose built cellular container-ships. The time spent by the author working with the project sponsors allowed familiarisation with the processes of loading and unloading containerised cargo. This domain familiarisation process, coupled with discussions about general problem features with maritime personnel

[6,12,68], led to an attempt to model the relationship between container-ships' cargo space and containerised cargo by using the entity modelling methodology. [42] Entity relationship modelling involves taking a written description of a physical pattern and extracting the entities, or 'units', which seem most important in the description. The relationship between each pair of entities is then determined. In this case, the physical pattern is the cargo-space and cargo of a container-ship. Entity modelling greatly facilitates problem conceptualisation, partly due to the ease with which models can be modified as and when relationships become better understood. The resultant entity-relationship-model generated by this problem is described in Appendix A.

The entity relationship model was used to generate a physical data-structure, suitable for experimentation with general search algorithms. The next section discusses how search was applied to an abstraction of the container-ship stowage problem.

7.4 Search applied to loading an abstract container-ship

Other authors have assumed that it is reasonable to employ a one-to-one relationship between standardised cargo and stowage locations. Initially, it was envisaged that the model used by this author to solve the container-ship stowage problem would also be based on this assumption. This section illustrates the lessons learnt from experimentation with the physical implementation of such a model. In particular, this section describes this first prototype implementation used in this project, and the consequences arising from applying search algorithms to the prototype.

7.4.1 Development of a container-ship abstraction

The standard container address system used by stowage planners was outlined in section 2.3. An early attempt to develop a suitable data-structure for modelling the cargo-space of a typical cellular container-ship involved exploiting that system. At this stage in the research process two simplifications to the container-ship stowage problem were made. Firstly, it was assumed that a one-to-one relationship between stowage locations and containers would prove useful in solving the problem. Here, a representation of the cargo-space as a set of individual slots that contain single items of cargo was used. This representation was based upon the work of Shields [13] and Botter [36] (outlined in Section 5.2.2 and Section 5.4.1). Secondly, it was assumed that reasonable intact stability could be maintained through the use of ballast. This simplification was assumed reasonable due to descriptions given by Sato *et al* [46] (described in Section 5.6). Therefore, at this stage in the development process the container-ship was conceptually reduced to that of a simplified cargo-space where relationships between stowage locations could be determined, mathematically, by referencing the container address. The first simplification made above was subsequently found to restrict the usefulness of generated solutions (as will be explained in Chapter 8). It was necessary to encode, initially, a simple abstraction of a cellular container-ship, which could be used to explore the simplifications of design and implementation decisions. The abstraction chosen was the model of a container-ship commonly called a *Box Barge* (illustrated in Figure 7-1). The Box Barge is a much scaled down model of a container-ship that retains all important characteristics such as holds, bays, hatch-lids, and stowage-locations, but without such considerations as bulkheads, crew quarters and ballast tanks.

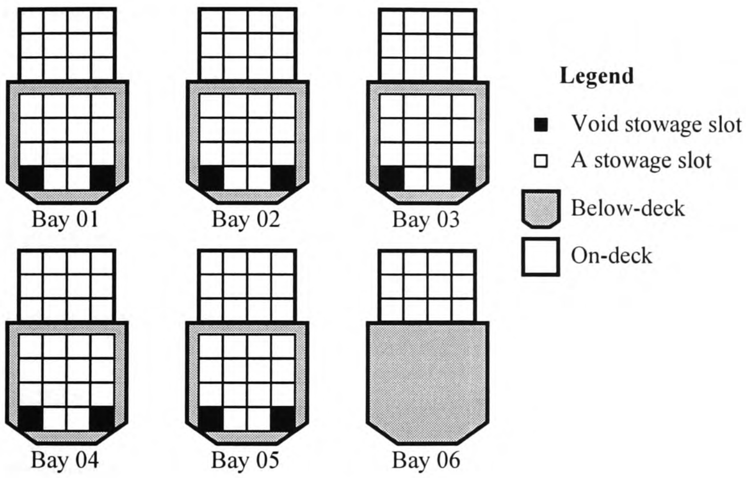


Figure 7-1 Cross-section of the Box-Barge

The Box Barge representation of a cargo-space was based upon a matrix (see Figure 7-2). Experimentation was then performed with search algorithms, where the goal was to place generalised, containerised cargo into the cargo-space. Each container had associated with it a destination, weight and class (where the class reflected segregation requirements).

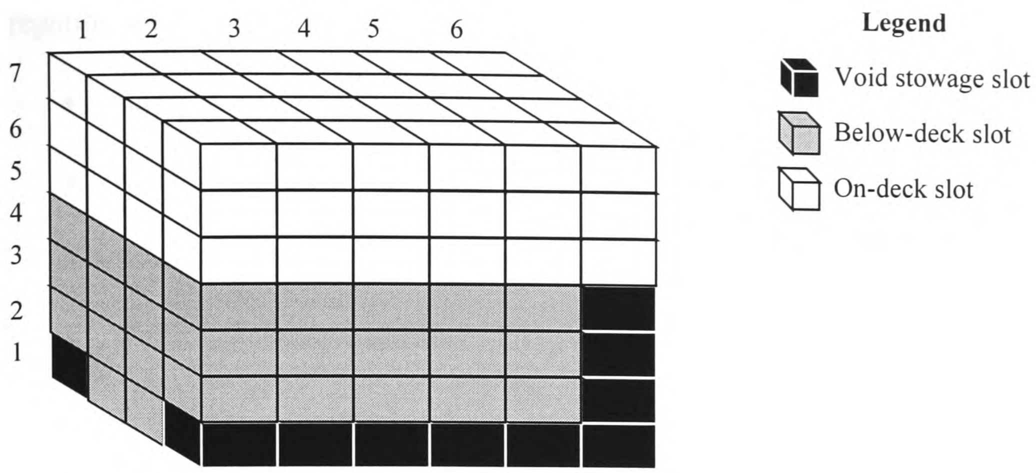


Figure 7-2 Matrix representation of the Box Barge

Experimentation with general search algorithms was limited to generating a stowage pattern for a current 'port-of-call'. The criteria (taken from the general criteria identified in Section 3.4) used to establish an optimum stowage configuration were:

- that heavier containers were to be stowed lower than lighter containers;
- that containers were to be stowed from the centre of the Box Barge outwards;
- that containers with the furthest to travel are stowed at the bottom of stacks;
- that containers with the same destination should, where possible, be stacked together;
- that containers with the same destination should be stowed in the same bay.

In addition to these criteria were constraints upon placement. These included all containers requiring support (either by the Box Barge casing, hatch-lid or another container), and requiring segregation of 'hazardous' types. Four levels of segregation were included, namely:

- adjacent;
- within the same compartment;
- within an adjacent compartment;
- within two compartments.

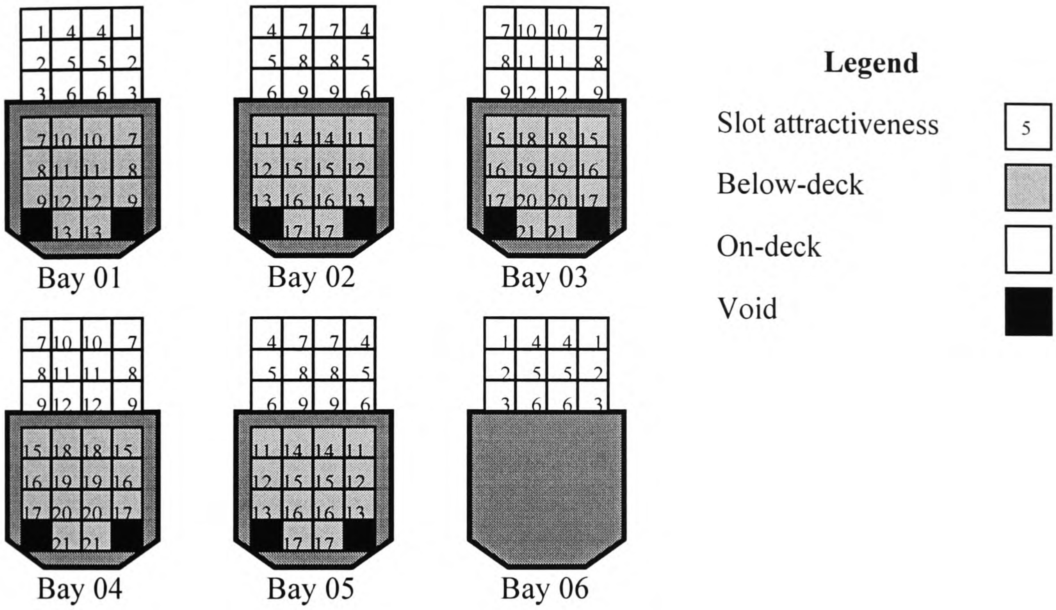


Figure 7-3 Box Barge showing example slot attractiveness values

An evaluation function (see Chapter 4) was developed from these criteria which used an assigned 'attractiveness' for each filled slot in a stowage pattern to determine the quality of that pattern (the attractiveness values of filled slots being totalled to determine the quality of the overall distribution of cargo). Therefore, placements of containers in slots with high attractiveness values is encouraged (an illustration of slot attractiveness values is given in Figure 7-3). Combining the weight of the containers with the attractiveness of the slot within the evaluation function encouraged the placement of heavier containers lower down and towards the centre of the cargo-space. The quality of the stowage configuration is further defined by analysing the distribution, by destination, of containers throughout the bays and stacks. For example, an ideal stowage for a heavy container would be low down in a central bay and stack, and to have other containers with the same destination stacked with it. The quality of the stowage pattern is further defined by penalising solutions where over-stowage (described in Section 3.2.2) has taken place. In this way, it was envisaged that optimal stowage solutions could be generated where over-stowage was minimised, containers were distributed from the centre out, with heavier containers stowed lower than lighter ones, where good block stowage could be accomplished and all constraints were satisfied.

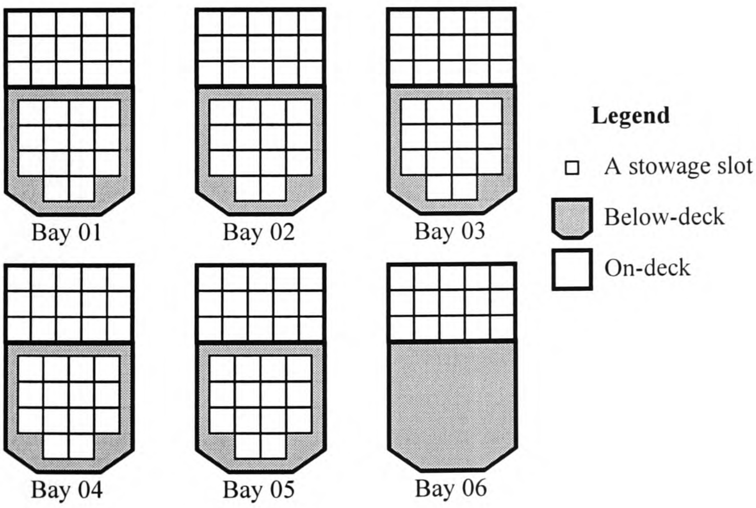


Figure 7-4 Example of a Box Barge's cargo-space

The matrix representation for the Box Barge was developed further in order that a broader range of cargo-space configurations could be represented. Figure 7-4 illustrates a configuration where more lateral stacks exist above deck than below. This type of configuration is important as it gives rise to the *inter-hatch stack* (which will be discussed in the next section in more detail).

The model developed so far equates bays to regular, simple (basically cubic) spaces bounded by exactly six faces. Modelling a container-ship in this simple way results in a large number of void stowage locations due to the irregular geometry of a typical container-ship's cargo-space. An example of typical variation in cargo-space geometry is shown in Figure 7-5. In the implementation of search algorithms, these void spaces would be needlessly processed.

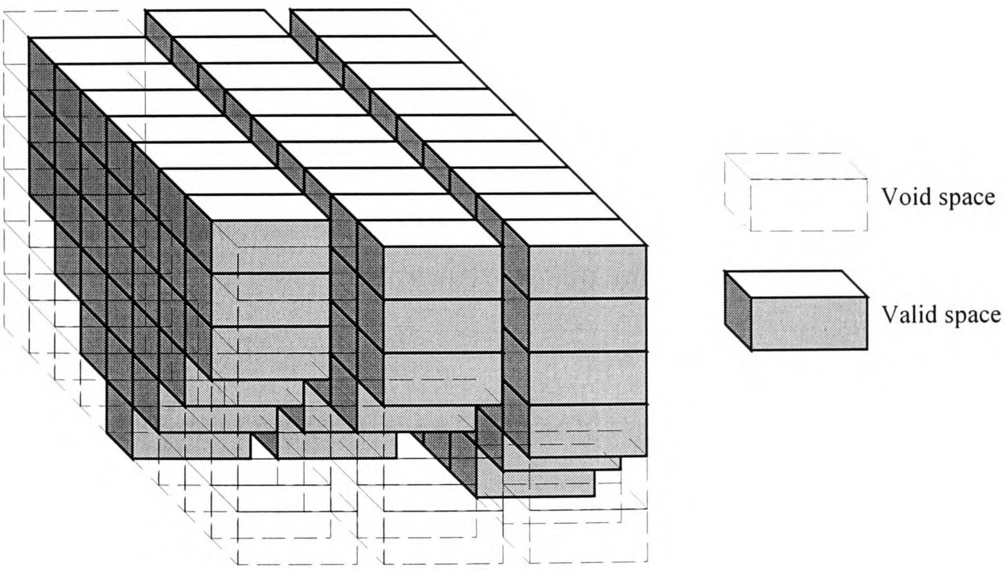


Figure 7-5 Sample cargo-space showing void matrix spaces

Only modelling the actual cargo-space reduces the computational size of the problem by removing the need to process void stowage locations. This is accomplished by replacing the matrix model with a model made up of linked entities (the design for which is shown in Figure 7-6).

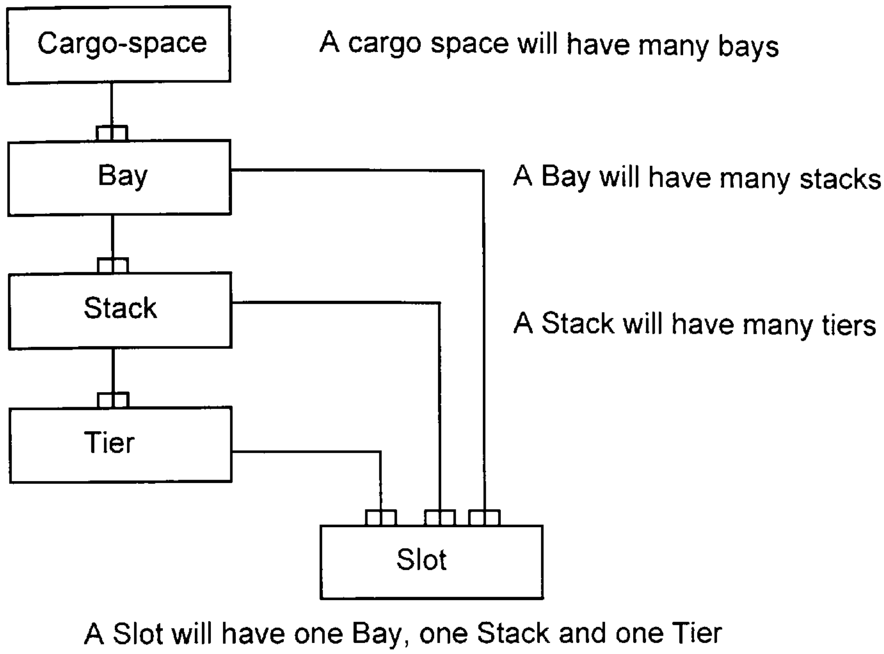


Figure 7-6 Entity relationship model for a cellular cargo-space

The design shown in Figure 7-6 was physically implemented using linked-lists, as shown in Figure 7-7. The linked list structure that used to model the cargo-space (C) is made up of a lists of bays (B), stacks (S) and tiers (T). The tier is associated with a stack and each stack is associated with a bay. Each slot within the cargo-space is made up of a bay-stack-tier combination (as illustrated in Figure 7-7).

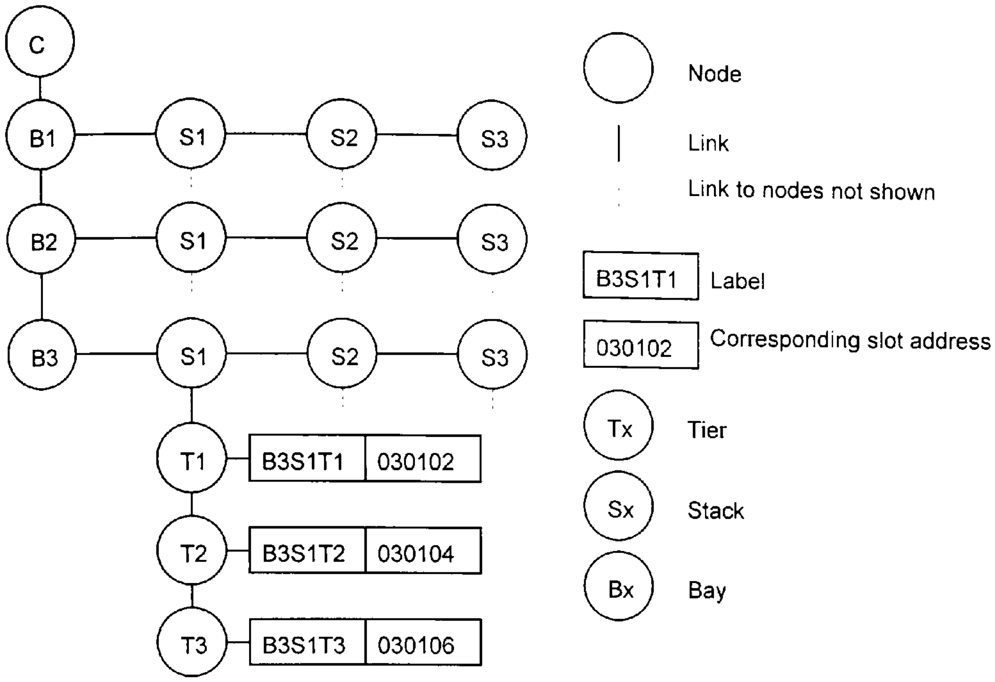


Figure 7-7 Diagram showing linked list representation of a cargo-space

Representing the cargo-space as a linked-list eliminates the void spaces present within the matrix representation. However, whereas determining the relative position between slots is straight forward within the matrix representation, semantic information about the relative positions of slots within the linked list must now be introduced to facilitate processing requirements such as hazardous cargo segregation.

7.4.2 Modelling semantic relationships within the linked list

Physically implementing the container-ship model described above introduced a further requirement that relationships between slots (a slot is made up of one to many cells, described in Section 2.3.1, and is used here to described the physical cargo-space occupied by an item of cargo) and container-ship attributes (such as hatch-lids) also be modelled. These relationships are modelled using a *semantic network* [16], with each relationship being encoded in preparation for the search algorithms used in this approach to solving the planning problem. Before the actual encoding of these relationships is discussed in detail, an introduction to semantic networks follows.

Semantic networks are a commonly used diagrammatic method of modelling the relationships between entities. A semantic network is constructed to convey meaning to the reader. A semantic net is a diagrammatic representation in which, lexically, there are nodes denoting objects, links denoting relations between objects, and application-specific link labels. Structurally, nodes are connected to each other by labelled links. In diagrams, nodes appear as circles, ellipses, or rectangles, and links appear as arrows pointing from one node, the tail node, to another node, the head node. [*Ibid.*] Whereas entity relationship diagrams facilitate the generation of structures that contain data, semantic networks are used to provide descriptions for how entities interact with each other. Understanding how entities interact facilitates the generation of processes designed to manipulate the data structures revealed by an entity relationship modelling exercise.

A cross-section of a container ship contains many relationships. In order to implement and use a model of the container-ship, all of these relationships must be represented. Even seemingly obvious relations must be explicitly defined so that a computer may simulate interaction with the domain. For example, although the relationships between hatches, bays, stacks, tiers, hatch-lids and cargo is apparent to a human planner, these relationships must all included explicitly in a computer model to allow effective processing and generation of solutions.

A semantic network can be used as a transitional method (that the act of doing helps the programmer understand the domain) of representing a structure that is to be physically implemented in software. Simple relations, such as *is-adjacent-to* and *touches*, can easily be defined (illustrated in Figure 7-8, in which nodes are used to represent bays, and Figure 7-9, in which nodes are used to represent cells and hatches).

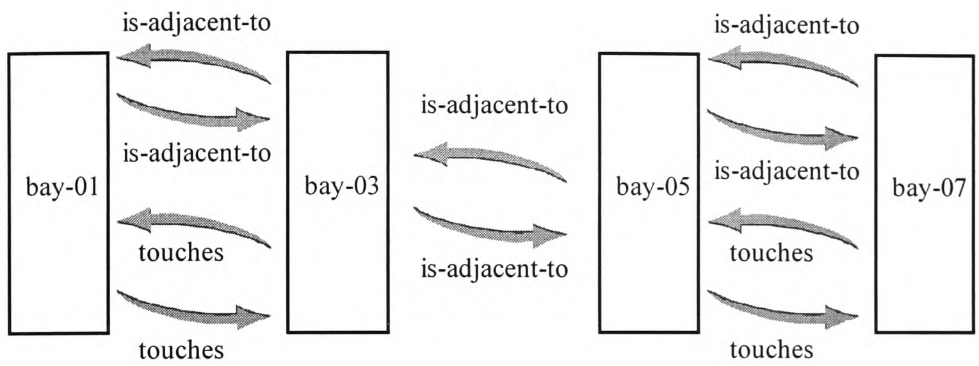


Figure 7-8 Example semantic network showing relationships between bays

The conceptual description of relationships using semantic networks assists both in further understanding the relations between entities, and in eventually physically implementing this semantic information. However, although physically encoding these relationships symbolically is comparatively simple, processing this stored information after construction can become computationally high when considering all relationships at a cellular level (Chapter 8 describes how experience with modelling the container-ship at a cellular level culminated in a fundamental revision of the container-ship model whilst retaining the principle relationships identified here.)

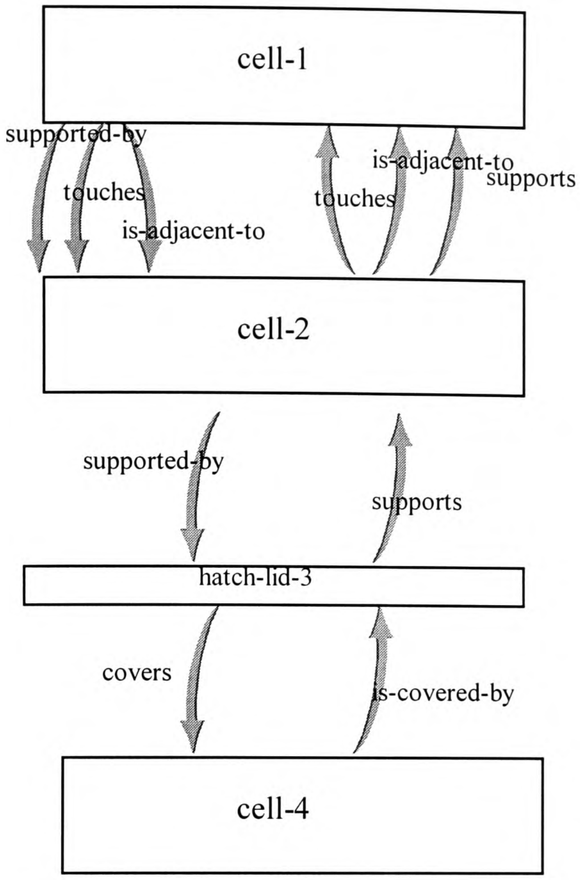


Figure 7-9 Relationships between cells and a hatch-lid

7.5 Experimentation with the cellular model

This section explains how the Box-Barge abstraction of the planning problem, described in the previous section, was implemented. The method of applying Search to this implementation is also described. Finally, the lessons learnt from this implementation are explained.

7.5.1 Implementing the abstract model

Encoding all the relationships between the components that make up the cargo-space of a container-ship (such as how bays, stacks, tiers, slots and hatch-lids relate to each other, described in Section 7.4) would be laborious, especially when these relations would have to be determined for each alternative container-ship. Therefore, based upon the theoretical model described in the previous sections, a computer program was developed that allows a user to construct graphically, a physical data-structure for the cargo-space of a cellular container-ship. From the physical data-structure the program automatically generates the required semantic relationships.

The logical model of the structure of a cellular container-ship's cargo-space described in Section 7.4.1 was updated to include links between slots and hatch-lids and physically implemented using linked-lists [79] (illustrated in Figure 7-10). This physical structure permits a variety of different cellular container-ships to be dynamically modelled and displayed.

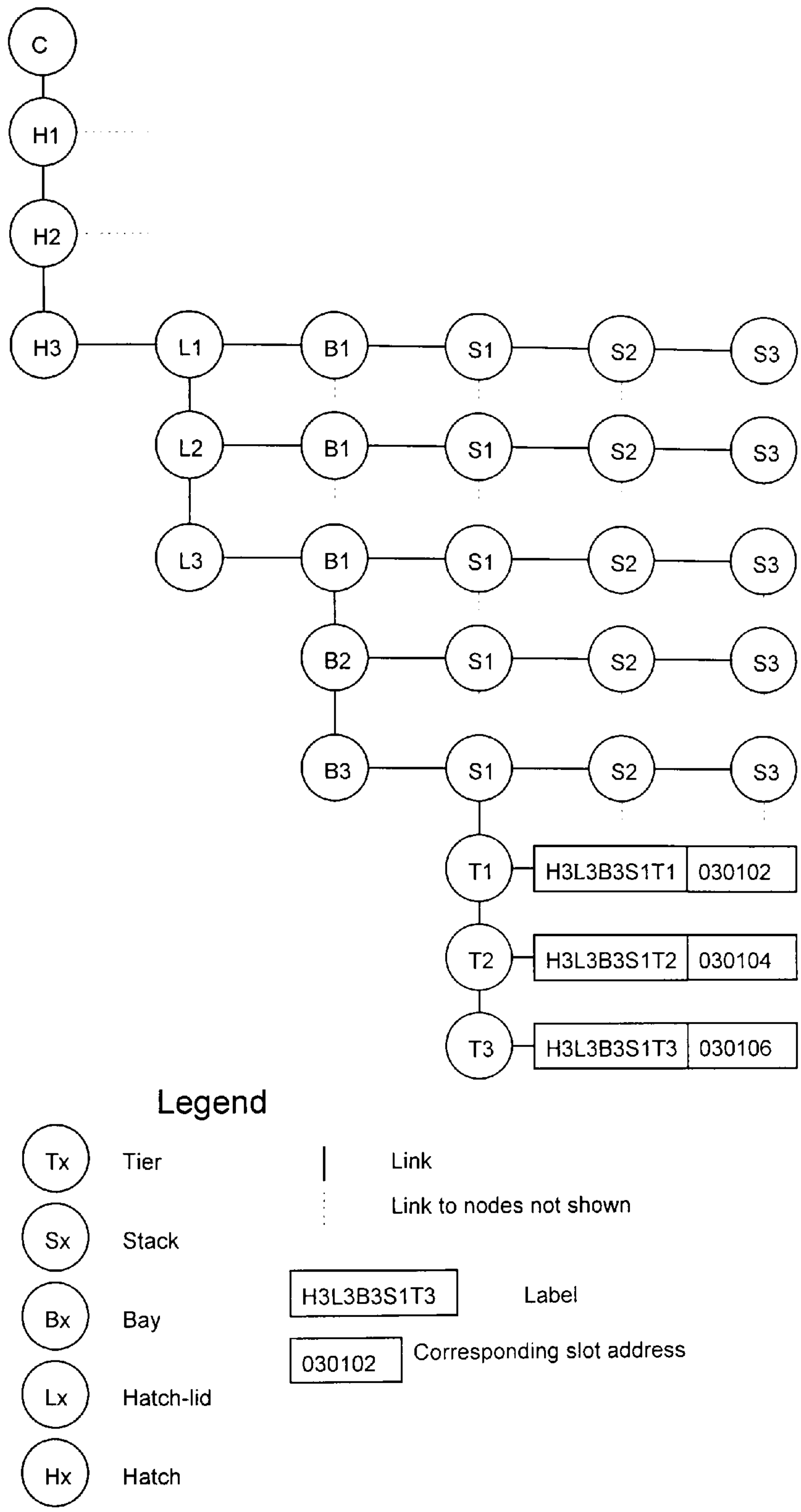


Figure 7-10 Updated cargo-space linked-list

Recursively parsing this linked-list enabled important relations between components of the cargo-space, such as *cell adjacency* and *cell supported-by*, to be captured in addition to the data-structure that represents each of the stowage locations (or slots) within the cargo-space. These data-structures are then stored in a format compatible with the planning prototype program (discussed in the next section).

The physical data-structures facilitated the application of AI problem solving techniques to the Box Barge abstraction of the planning problem. The importance of this tool can not be understated since capturing and creating the structure and relations by hand, given that vessels normally contain thousands of cells, would be a long and tedious exercise. The container-ship modelling utility enabled rapid construction of a cellular cargo-space data-structure with standard slot sizes (described in Section 2.3) and semantic relationships that captured the essence of the planning problem. After creating data-structures for container-ships of varying sizes with the modelling utility described above, a scaled down abstract model of a Box-Barge (illustrated in Figure 7-4) was generated. The physical data-structures for the Box Barge were produced in a format suitable for importing into the Franz Allegro Common Lisp the programming language used to encode the search algorithms applied to the Box-Barge loading problem.

7.5.2 Choice of programming language

For many years, search has remained the backbone of Artificial Intelligence. [16] The implementation of search applications led to the development of LISP (LISt Processing) as an AI programming language. [72] Although a powerful programming language, in the past LISP has failed to make an impact upon the computing community, partly due to the non-existence of a suitable PC based package. This lack of popularity may change with the introduction of Allegro LISP. The main strengths of LISP are its functionality, the ease with which symbolic structures can be represented and its use of recursion [15] to solve problems. These strengths make LISP less acceptable to the typical programmer who is not familiar with these practices. However, these same strengths make LISP ideal as a problem solving programming language, and it was chosen as the language for implementing the first prototype system in this project. Further details on the suitability of LISP as a problem solving language, and an example of its use, can be found in Appendix B.

7.5.3 Encoding the Box-Barge using LISP

The state-space (introduced in Section 4.1.1) that represents all possible loading configurations can be explored using the standard principles of problem solving [16] described in Appendix B. The following section describes the data-structures and search algorithm used to generate solutions to the Box-Barge abstraction.

This section describes how the cellular cargo-space was physically encoded. As explained in Section 7.3, the representation used to describe the cargo-space of the Box-Barge assumed that a one-to-one relationship between stowage locations and containers exists. Working under this assumption a source node for the state-space was defined as containing two parts:

- a representation of the cargo-space;
- a value that approximates the relative effectiveness of the solution so far (determined by the evaluation function).

The cargo-space representation was made up of a list of elements, with each element corresponding to a stowage location, for example:

(((cell-010102 nil) (cell-010104 nil) ... (cell-0504008) 0)

Each element within the cargo-space representation is made up of two parts: a location address (described in Section 2.3) and the location contents. Initially each location is empty, indicated by the 'nil' object (nil is used within LISP to represent an empty element) above for the contents. A loaded cell would be represented by, for example:

(cell-010102 (container-1 Hamburg Red 20))

In this example, the cell found at bay 1, stack 1 and tier 2 has a container destined for Hamburg, with a tokenised hazardous classification of 'Red' and a weight of 20 tonnes stowed there. Semantic relationships (discussed in 7.4.2) are similarly encoded in a second data-structure, namely:

- each slot has a list of adjacent slots associated with it (e.g. slot 010282 in Figure 7-11 has slots 010284, 010482, 010484, 010084 and 010082, in addition to the corresponding slots, not shown, in Bay 02, adjacent to it);

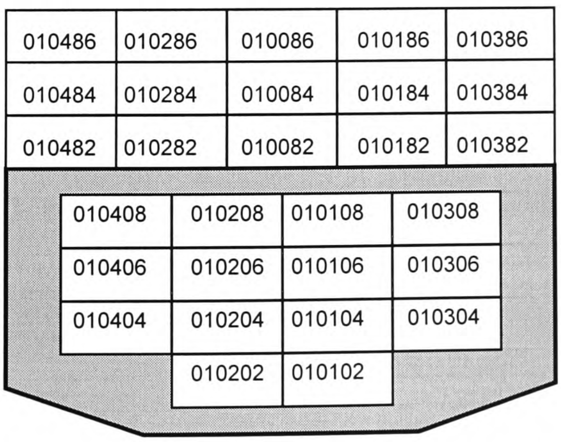


Figure 7-11 Slot address references

- each bay has a list of adjacent bays associated with it (*e.g.* bay 02 is adjacent to bays 01 and 03);
- each slot has an a supporting element (either another slot or part of the ship) associated with it.

Figure 7-12 illustrates the initial contents of the Box Barge cargo-space. A number of the cells have been pre-filled with containers to represent the state of the cargo space after discharge at its current port of call.

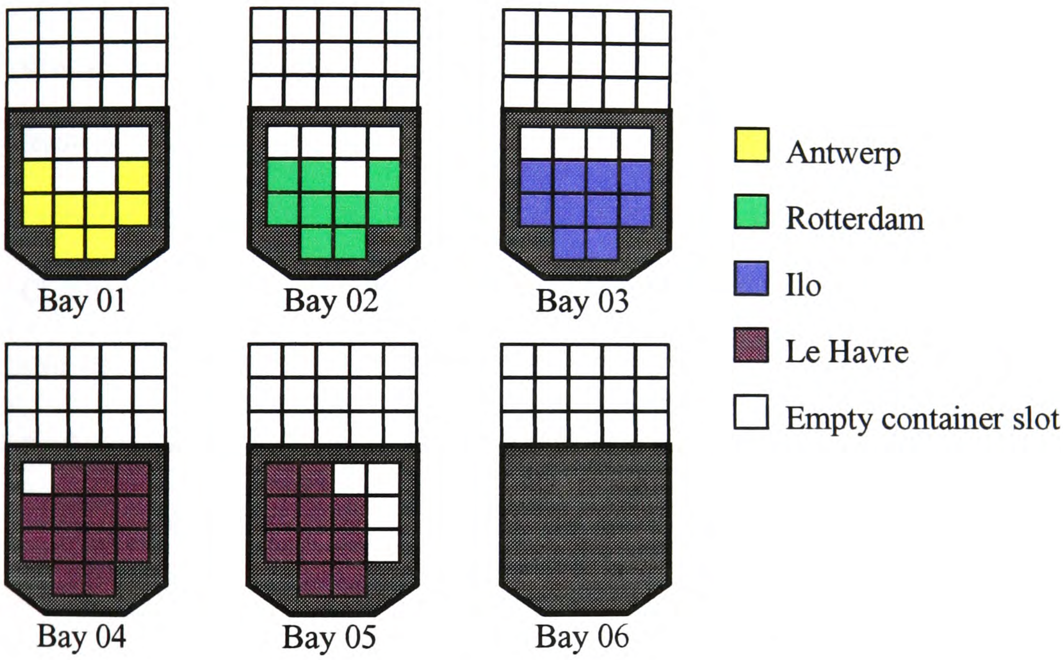


Figure 7-12 Box barge before loading

Several lists of containers were prepared for experimentation where each list item was is comprised of:

- a destination;
- a weight;
- a hazardous status (represented by tokens, such as 'Red');
- a container identifier.

Each generated load-list was pre-sorted by destination and weight. Search was applied to the Box-Barge where only the current port-of-call was considered. The object of the placement algorithm was to spread the weight of the load across the ship, starting from the centre, working outwards, whilst ensuring that illegal colour adjacencies (representing hazardous container segregation) did not take place and overstows were minimised.

7.5.4 Applying search to the Box-Barge abstraction

A stowage planning prototype program, using the implementation of the data-structures described in Section 7.4, was written. This prototype demonstrated the feasibility of the use of search algorithms, as will be detailed in this section. After experimentation with exhaustive search (described in Section 4.2.1), the Hill Climbing [16] search algorithm (described in Section 4.2.2) was applied to the data structures (*i.e.* the cargo-space and relationships) generated by the container-ship modelling tool described in Section 7.5.1. These structures and relations were combined with a list of abstract containers that were to be loaded into the box barge. The Box-Barge was partially pre-filled with a set of dummy containers in order that the overall model would better represent the state of a real container-ship when awaiting loading.

The Hill Climbing algorithm was selected as the primary experimentation algorithm since it demonstrates how an effective, if not optimal, stowage solution can be generated in a reasonable amount of time. (This is an implicit characteristic of the Hill Climbing algorithm that is evident within other 'search' based algorithms, such as Glover's Tabu Search described in Chapter 8, to solving optimisation problems.)

Before applying search, the list of containers was sequenced such that containers that were to travel the furthest would be loaded first. Within the destination prioritisation, heavier containers were selected first for loading. The evaluation function used in Hill Climbing to measure the attractiveness of a proposed stow incorporated the following components introduced earlier in Section 7.4.1, which are re-iterated here from an operational point of view:

- a solution is penalised for having lighter containers stowed below heavier containers;
- a solution that has containers stowed from the centre of the Box Barge outwards scores highly;
- a solution that has containers with further destinations stowed above containers with nearer destinations (an overstow) is penalised;
- a solution with stacks of containers with mixed destinations is penalised.

Colour was used to represent hazardous containers stowed within the abstract model to demonstrate how invalid stowage combinations could be removed from the available pool of solutions. The problem was simplified by reducing the large number of different hazardous cargo-types to four. The placement of different coloured containers in relation to each other was used as a constraint upon the solutions generated.

7.5.5 Results

Hill-climbing achieved good results and proved more suited to the problem when considering planning for a multi-port route. This is because an exhaustive examination of every possible stowage configuration over a multi-port voyage, even for the Box-Barge abstraction, is inordinately large. As shown in Chapter 4, Hill-Climbing greatly reduces the size of an explored state-space by pursuing only the most promising stowage configuration. Figure 7-13 illustrates an example of a resultant stowage configuration for a complete Box-Barge.

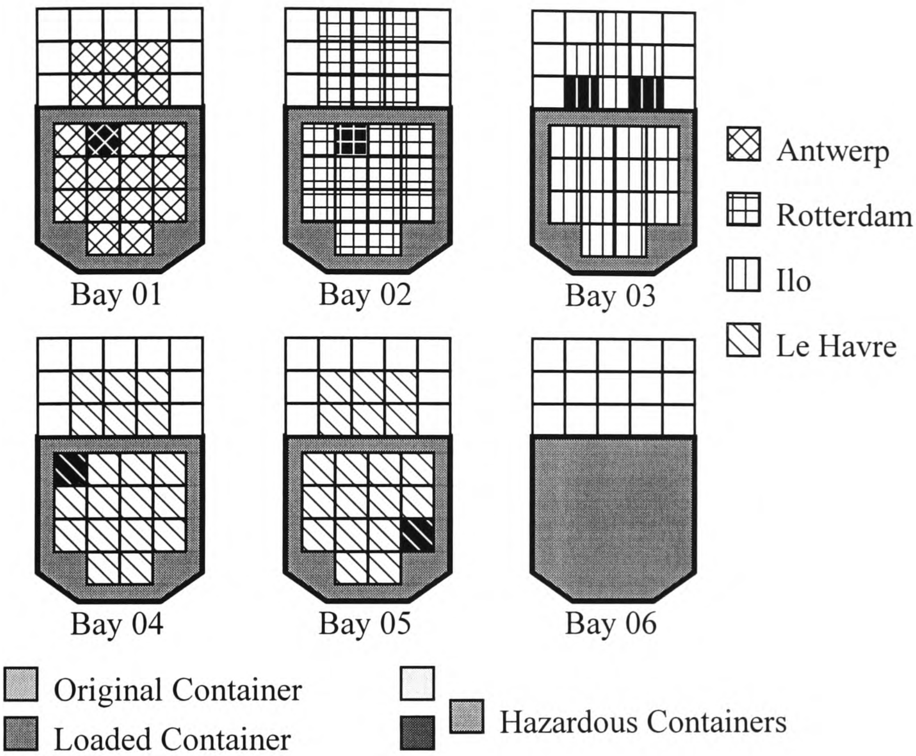


Figure 7-13 Box Barge after loading

The resultant configuration had no overstows, containers with like destination were blocked together in the same bay and hazardous segregation was maintained.

7.5.6 Conclusions drawn from the Box-Barge abstraction

The main objectives of demonstrating how a container-ship can be represented and loaded using general search theory have been met. However, attempts to integrate out-of-gauge containers into this abstraction of the cargo-space uncovered the main weakness of this representation, that of the one-to-one allocation of containers to stowage locations. Further investigation revealed a many-to-many relationship between cargo items and standard slots (*i.e.* a slot can hold multiple items of cargo, such as flat-racks, and an item of cargo may occupy many stowage locations [6, 12]).

In addition to the above consideration, there is another problem concerning the multi-port nature of the stowage problem. Attempting to make individual placements of containers across a multi-port voyage (illustrated in Figure 7-14), whilst offering an opportunity to generate optimum solutions, is not practical because of the combinatorial and computational aspect of the problem. [36] Figure 7-14 illustrates this aspect: each proposed solution at Hamburg requires consideration and a corresponding proposal for leaving Antwerp - the number of proposed solutions which must be explored rises dramatically.

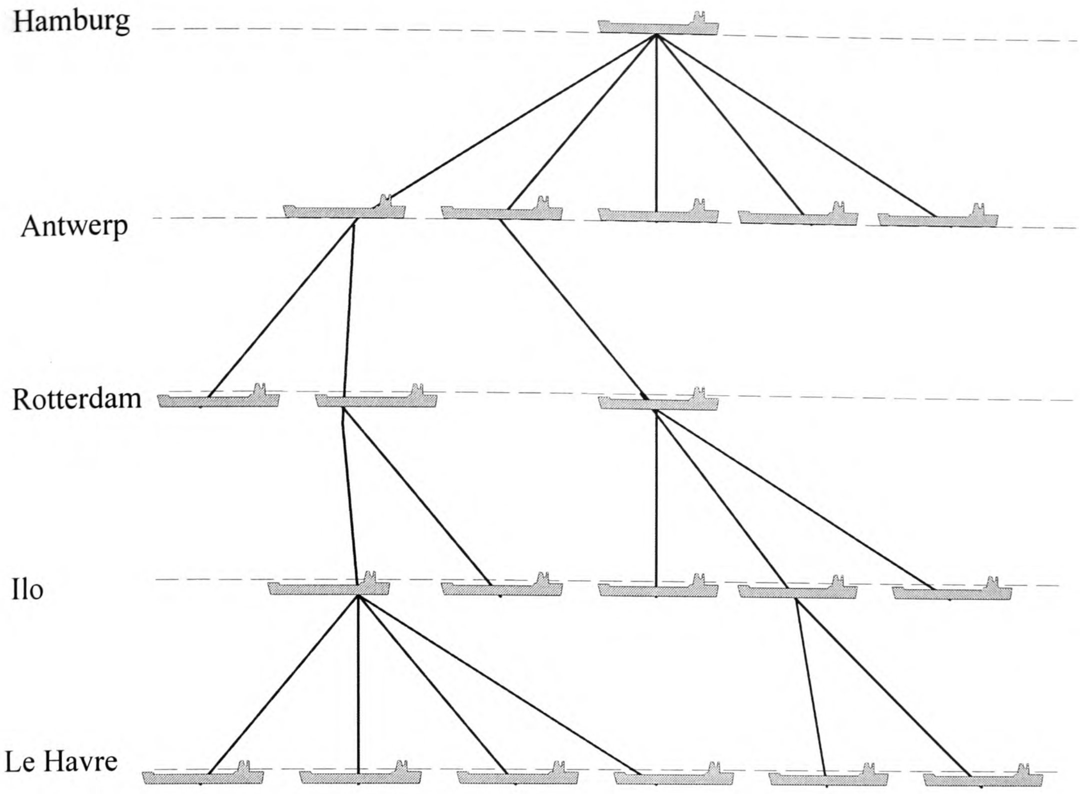


Figure 7-14 Example search tree showing branching during a multi-port voyage

This initial experimentation highlighted the importance of modelling the processes undertaken by the human planner, namely: that the planner makes no attempt to plan each individual container placement, but instead adopts a two part process of generalised and specialised planning (strategic and tactical planning described in Section 3.5);

The simplification processes undertaken, although encountered in other authors' work, fail to represent the complex relationship between a cargo-space and cargo. Complications centre on the various sizes and types of cargo carried (described in Section 2.1.3). Whereas the above prototype deals with the issue of hazardous cargo segregation, other requirements such as access to external power-sources, must be built into the model. The placement of containers that do not correspond to the cellular layout of a container-ship's cargo-space, although discounted by other researchers, is an important consideration that must also be dealt with.

The complications associated with dealing with containers of different dimensions point towards a move away from a the one-to-one, container-to-slot model to a more abstract many-to-many model. Viewing the cargo-space as a collection of areas to fill would better model the how the human planner views the cargo-space. A volume based approach to strategic planning, where each container has an associated TEU value is described in the next chapter.

8 AUTOMATING PLANNING

8.1 Introduction

Chapter 7 outlined initial experimentation with an abstraction of the container stowage problem where specific container to slot placements were considered. The literature survey, and in particular the work of authors Shields [13] and Botter [36], indicated that an approach centred on the placements of particular instances of expected container types to specific slots would be appropriate. However, the lessons learnt from the experimentation described in Chapter 7 show that the search space for such an approach is prohibitively high, preventing examination of a reasonable number of alternative stowage configurations in a reasonable length of time.

In this Chapter, the task of loading containers is divided into two main stages, in line with processes used by human planners (described in Section 3.5), termed *strategic* and *tactical* planning by this author. The following sections will show how the design of this approach evolved, and will explain how this approach allows for consideration of more stowage configurations in a reasonable length of time.

8.2 Automating the strategic planning phase

The author strongly conjectures that in the strategic planning phase (described in Section 3.5), a human planner considers placing containers into approximate positions in the cargo-space rather than in specific cell locations. An analysis of the documents used by human planners (the General Arrangement, Outline Plan and Bay Plan described in Section 3.3.2) revealed three conceptual levels of planning, from the general (or strategic, described in Section 6.2.3.1) to the specific (or tactical, described in Section 6.2.3.2). In this section, the author explains the development of a new approach to computerised planning, based upon *blocking*. This approach uses an abstraction of a cargo-space designed to model the processes used by human planners in the strategic planning phase.

8.2.1 The advantages of abstracting a container-ship's cargo-space

In the work described in Chapter 7 it was assumed that an appropriate approach to solving the container-ship stowage problem was to generate a one-to-one relationship between expected containers and specific stowage locations. That is, a specific location would be chosen to hold one of a group of expected containers of particular type and weight (see Section 7.3). The solution space for this approach is large, as there would be many slots that would be appropriate for stowing each grouped type of containers. The solution space can be reduced by using an abstraction of the cargo-space, allowing groups of containers to be assigned to spaces that are less specific than precise slot locations.

Applying search to an abstraction of a container-ship's cargo-space would have the following advantages:

- the need to load containers individually would be removed. Instead, containers with similar characteristics could be grouped together and loaded in a collection of cells (or *block*). The available alternatives for placing containers would be reduced, generally, from hundreds to only as many distinct areas as the cargo-space is abstracted into;
- the number of general stowage configurations that can be considered in a reasonable length of time would be greatly increased;
- The use of groups of generalised containers, rather than individual containers, also models well the way in which human planners use the statistical data on containers with which they are provided.

During the development of these abstractions it became increasingly clear that modelling how a planner decomposes the stowage planning problem into smaller sub-problems, rather than applying advanced search theory to an unrepresentative simplification of the problem, would facilitate finding a computer-based solution. The following section describes the development stages that led to the final 'blocked' cargo-space abstraction.

8.2.2 Developing the cargo-space abstraction

This section describes how a suitable representation for the cargo-space of a container-ship evolved as the author's understanding of the planner's perceptions and processes increased. The intent was to find a representation for the cargo-space that reduced the combinatorial and computational complexity of the strategic stowage planning process.

8.2.2.1 Cargo-space representation as a set of stacks

A summary of stowage objectives (fully described in Section 3.4) during the strategic planning phase are:

- to maximise cargo-space usage;
- to maximise crane usage;
- to minimise hatch-lid movement;
- to minimise over-stowage;

whilst ensuring that all constraints relating to intact stability are met.

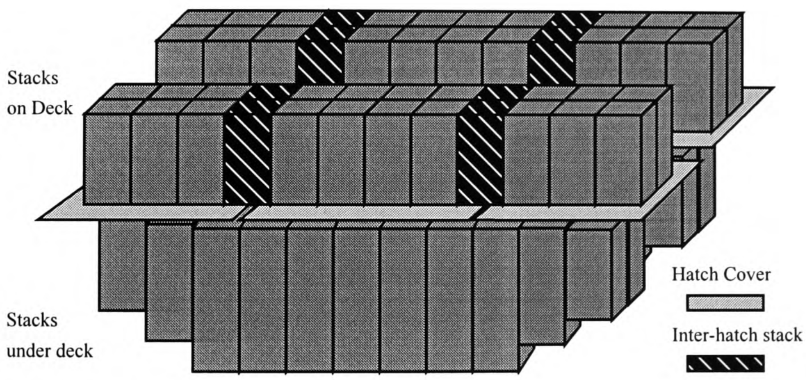


Figure 8-1 Cargo-space represented as stacks and hatch-lids

The cargo-space could be represented as a set of stacks that relate to hatch-lids (illustrated in Figure 8-1), where each stack is 'filled' with containers. This would clearly be an abstraction that reduces the number of options needed to be considered, at each stage of search, by removing the need to make individual allocations of containers to slots. Containers could instead be allocated, at this stage, to stacks.

However, it is necessary to ascertain whether an abstraction allows the above stowage objectives to be met (or rather checked) in addition to ensuring that it reduces the search space.

Cargo-space usage can be maximised by ensuring that stacks below hatch-lids are used efficiently and any out-of-gauge containers are placed at the top of stacks (minimising any resulting void spaces). Efficient use of cranes can be arranged as easily when considering stacks as when all slots are considered. Consideration of cargo placement in relation to hatch-lids is as efficient when dealing with stacks as slots. Since the locations of containers within stacks can be determined later, over-stowage can therefore be minimised. It can, therefore, be seen that all of the objectives identified can be checked only by consideration of which stack each container belongs, regardless of which exact slot is allocated.

8.2.2.2 Cargo-space representation as a set of blocks

Although the state-space associated with the strategic planning phase can be significantly reduced by representing the cargo-space as a set of stacks that relate to hatch-lids, the state-space is still unacceptably large when considering a multi-destination voyage. Therefore, it was determined that the model of the container-ship's cargo-space used during the strategic planning phase requires further abstraction.

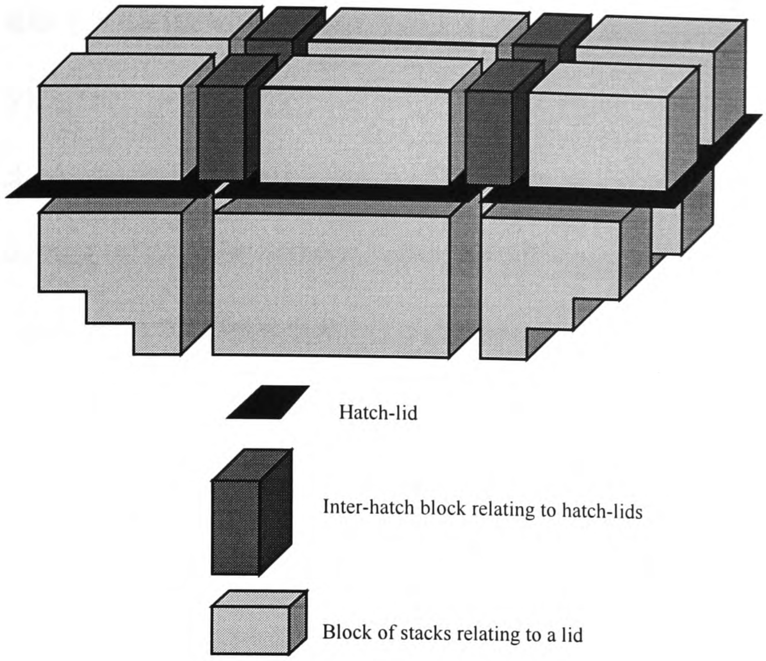


Figure 8-2 Example of cargo-space blocking

Blocking stacks of cells that share a common relationship to a hatch-lid (as illustrated in Figure 8-2) would still allow checking of the objectives given in Section 8.2.2.1. This abstraction also facilitates an approach to strategic stowage planning that more closely models the process undertaken by a human planner. The level of information about future cargo available, the general guidelines followed and the documents used (*i.e.* the General Arrangement and Outline Plan) by the human planner indicate a multi-level conceptual approach to planning stowage configurations.

In addition to allowing the stowage objectives specified earlier to be checked, blocking stacks together has the following advantages:

- the number of stowage locations considered when placing a container is reduced from the number of slots (2235 in the case of the Sirius [63] and 2169 in the case of Resolution Bay [64]) to the number of conceptual blocks (79 in the case of the Sirius and 86 in the case of the Resolution Bay);

- each cargo-space block has a TEU (or volume) capacity, therefore assigning a suitable volume value to all containers to be loaded will accommodate all dimensions of containers (including any out-of-gauge);
- constraints upon stowage of different types of container (such as reefer) and cargo (such as types that require specially treated cargo-spaces) into a specific block can be easily attributed by the assignment of properties which dictate the type of cargo that can be accepted there;
- intact stress and stability can be calculated for the abstract model to an acceptable degree using an approach based upon the work of Sato *et al*[46];
- this abstraction better models the placement of 'generic' containers (the vague container details provided by upon statistical forecast data);
- The modelling of semantic relationships attributed to a container-ship becomes easier since far fewer relationships (cargo-space block as opposed to cell relationships) have to be determined.

The *blocked* cargo-space of the Resolution Bay container-ship is illustrated in Figure 8-3. Special attention should be paid to the concept of an *inter-hatch-lid* where three or more blocks are influenced by the same hatch-lid (such as blocks 78, 79 and 83). This is important since grouping containers with the same destination and placing them into blocks in such a way that a minimum number of hatches are affected during the loading and unloading process (described in Section 8.2.2.4) is one of the human planner's primary objectives (given in Section 8.2.2.1).

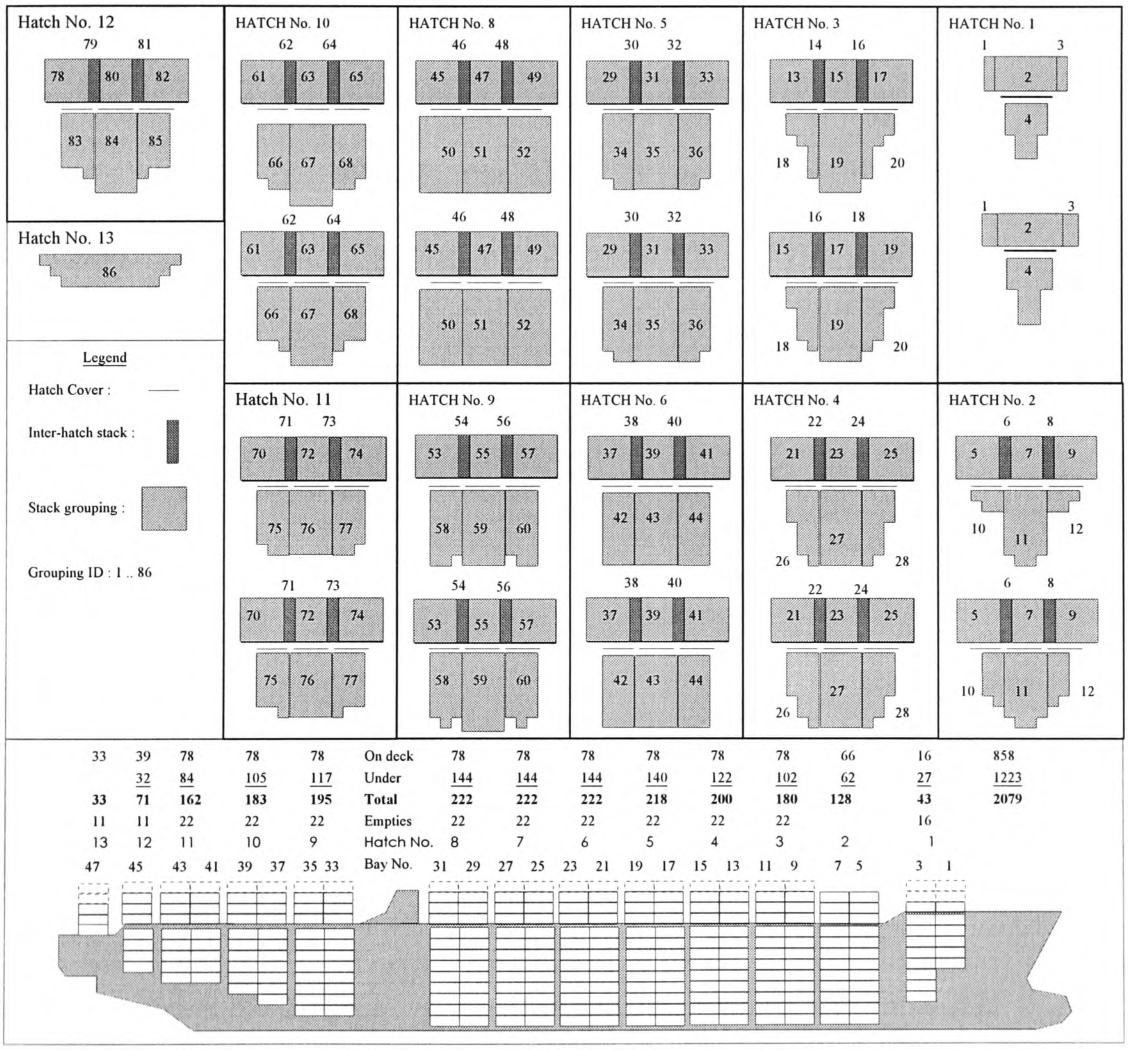


Figure 8-3 Outline plan of the Resolution Bay showing the 'blocks'

These blocks of stacks are spatial structures to be filled with containers. This conceptual view-point of the overall stowage space can easily be modelled using the a variation of the structures detailed in Section 7.4.1 where the object is now to reproduce the relations between blocks and the container-ship instead of stowage locations and the container-ship.

This relationship between blocks and hatches (illustrated in Figure 8-4) is important since care must be taken when placing containers so that hatch-lid movement is kept to a minimum (described in Section 3.3). (The implementation of this abstraction, illustrated in Figure 8-4, would have to return both A and B as the blocks which are *on-top-of* hatch-lid (i), but only one block, G, as being *below-hatch-lid* (ii) for example.)

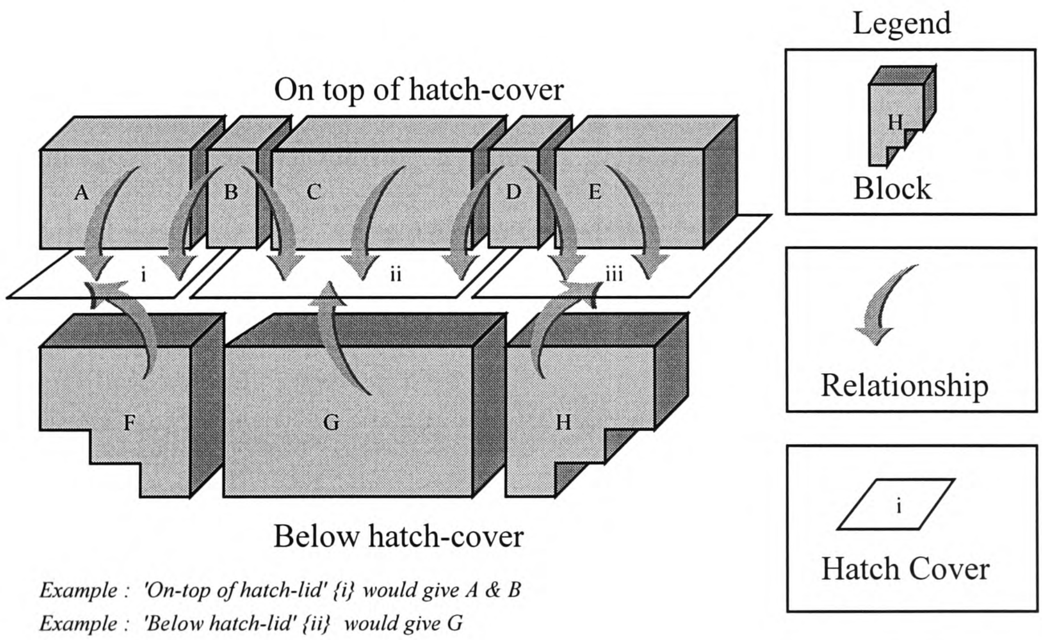


Figure 8-4 Semantic relationships between blocks and hatch-lids

Further analysis of the processes [68] and documents (described in Section 3.3.2) used by human planners during the strategic planning process led to further abstraction of the cargo-space. Section 6.2.2 described how long-term planning is first performed using the General Arrangement document and then finished using the Outline (or Letter) Plan. This led the author to further decompose the processes involved in producing pre-plans, such that it is reasonable to consider the placement of cargo, when using this abstraction, in two stages. Firstly, the stowage location for a container can be specified by its longitudinal position. Secondly, the location can be made more specific by choosing its latitudinal position.

8.2.2.3 Longitudinal blocking of the cargo-space

Blocking cargo longitudinally by hatch means that a location of a container is specified only by hatch-lid (*i.e.* as being either above or below a particular hatch-lid). An example is shown in Figure 8-5 for the Sirius [63] container-ship; longitudinal blocking here means specifying a number between 1 and 12. Within each longitudinal description of the space, there are several blocks to which the container could be assigned. This further specification of location is provided by a latitudinal assignment (described in Section 8.2.2.4).

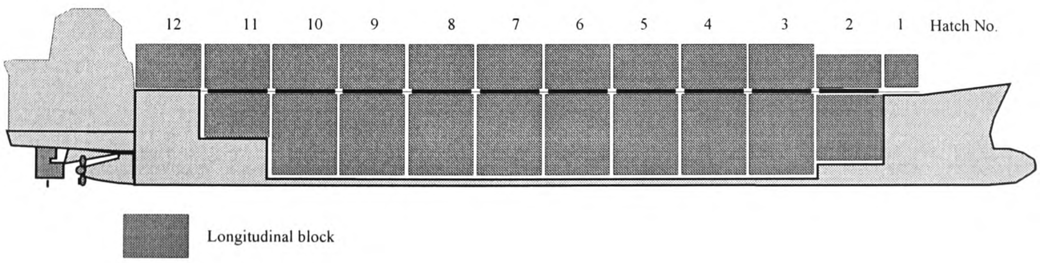


Figure 8-5 General Arrangement *showing longitudinal blocks*

This incomplete longitudinal specification is sufficient to ensure that crane usage is maximised since cargo can be spread across the ship appropriately (more specifically, cargo is split into as many hatches as the destination port has cranes with adequate spacing between hatches to allow the cranes to operate simultaneously, examples of how cargo is gradually distributed according to destination are given in Figure 8-6).

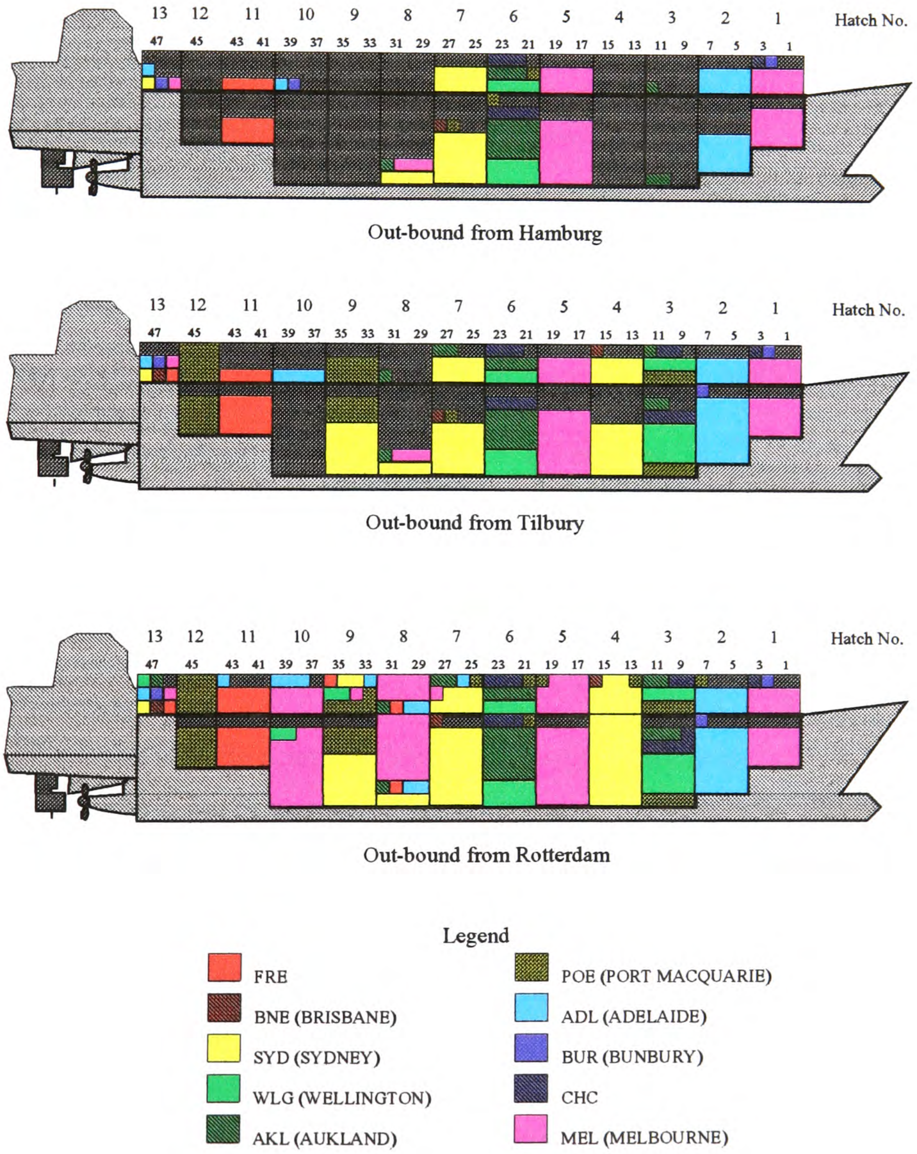


Figure 8-6 Example of a general arrangement showing blocked cargo

Representing the cargo-space in the above way reduces the stowage problem to that of simply distributing containers between a number of hatches. This spatial specification is still sufficient to allow the bending and trim constraints described in Section 2.3.6.2 to be calculated to a reasonable degree of accuracy (sufficient, at least, to give a good indication given the large amount of statistical information that will be included within the plan; it should be recalled that much of the cargo information being used is only statistical, and therefore not sufficiently detailed for exact calculations) for the resultant stowage configurations.

8.2.2.4 Latitudinal blocking of the cargo-space

The longitudinal blocking described above determines a good distribution of containers between hatches. Given the containers allocated to each longitudinal block the planning process can now move to considering which block within each hatch the containers should be placed in (illustrated in Figure 8-7). That is, it becomes necessary to distribute the containers between the latitudinal blocks associated with each longitudinal block. Only the containers within a particular longitudinal block are allocated to its corresponding latitudinal blocks. The state-space is restricted to the combinations of stowage patterns associated with the blocks within a single hatch.

With both longitudinal and latitudinal blocking considered, the locations of containers have now been specified to the degree of exactness as was described in the 'blocking' cargo-space abstraction of Section 8.2.2.2. Stress constraints relating to lateral distribution of weight and deck weight limits can be calculated to an acceptable level of tolerance (given the large amount of statistical information used at this stage in the pre-planning process) using this model. Planning for balanced weight distribution across hatches in this way will facilitate bringing the vessel to an even keel.

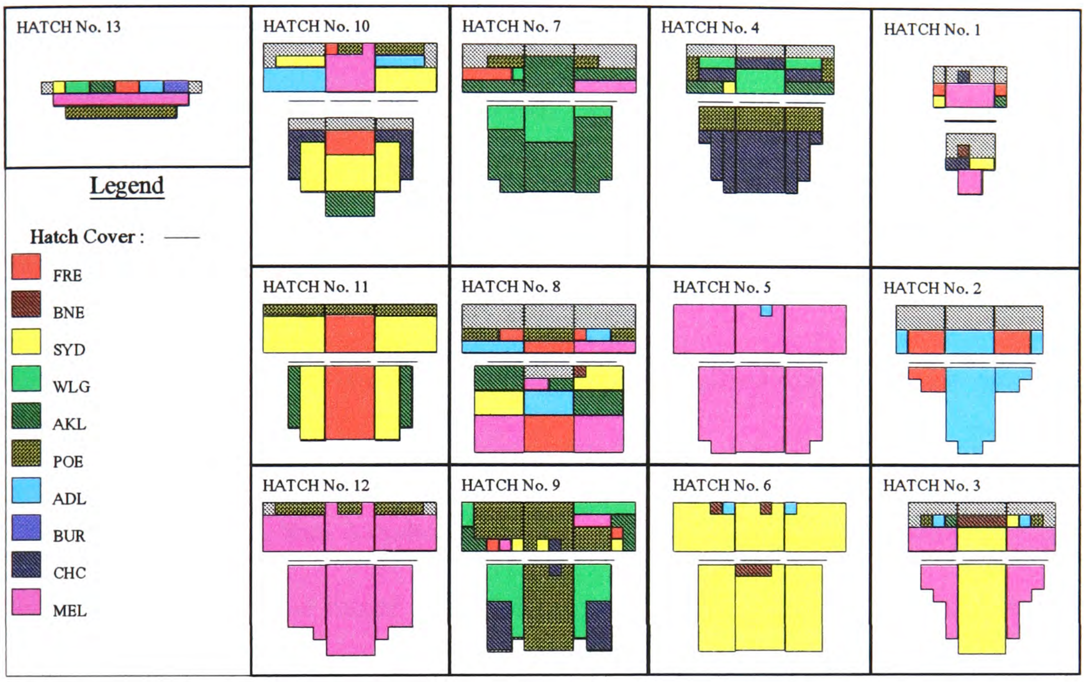


Figure 8-7 Generalised blocked Outline Plan for the Australian Venture

The blocked Outline Plan shown in Figure 8-7 [64] shows the cargo-space blocks associated with each hatch in relation to each of the hatch-lids.

8.2.3 Applying search to the strategic stowage problem

This section outlines how general search techniques can be applied to the strategic planning part of the container-ship stowage problem, given the abstraction process explained above.

The vast combinatorial problem described in Chapter 3 has been reduced in size by analysing the processes undertaken by the human planner. Each of the four stowage objectives outlined in Section 8.2.2.1 can now be met by applying standard search techniques to the longitudinal and latitudinal blocking processes described in Section 8.2.2.2. Search is performed through application firstly to a longitudinal blocking description of cargo-space content, and secondly to the refined latitudinal description.

8.2.3.1 Applying search to the longitudinal abstraction

The specific objectives for solving the container-ship stowage problem at a longitudinal level are stated in Section 8.2.2.2. To recap, they are that crane usage is to be maximised, that constraints, such as trim and bending moments are to be kept to within a tolerable range and special cargo handling requirements associated with containers, such as container type and content restrictions are met. Section 8.2.2 described a logical data-structure intended to facilitate the application of search. To apply search to the problem, a suitable physical data-structure that models the longitudinal cargo-space is required. An appropriate physical data-structure is described in this section.

The General Arrangement illustrated in Figure 8-5, where the cargo-space is divided longitudinally into twelve blocks, could be modelled physically in the same way as the Box-Barge described in Section 7.5.3. with the difference that certain properties associated with the cargo-spaces within each hatch would have to be included.

The properties that would need to be associated with each hatch are:

- that any restrictions on length of container (usually below-deck) be encoded;
- that the number of external power-supplies for powered containers be noted;
- that limitations upon the types of cargo that can be placed in each hatch be encoded;
- the maximum TEU containers that can be stowed above and below deck;
- any other specialised information specific to the particular container-ship be included.

With these properties incorporated into the Box-Barge data-structure, the algorithm for performing search would be constructed in the following way. Firstly, a load-list would be generated that, taking advantage of the generalised nature of the container descriptions in the statistical data used, would be grouped into *classes* of containers. This process follows the description given in Section 5.2.2 due to Shields. [13] Each class would be made up of containers that share the same characteristics. For example, all 20' containers destined for Hamburg would be placed in the same class.

Each of these classes would then be placed into ascending order of weight, typically with there being more than one container having with a statistically forecasted weight value (with these forecasts already playing an important part in shipping operators planning procedures) weight value. Initially, containers would be placed sequentially, using an appropriate search algorithm, with the result that a number (dependant upon the search algorithm implemented) of different stowage configuration combinations would be generated.

Where classes are made up of a large quantity of containers with large spaces within the cargo-space to place them into, more than one container would instead be placed in one go (the number of containers would be determined experimentally). The number of containers placed could conceivably start large and be reduced at each branch of the search tree (using, for example, a *simulated annealing* [69] algorithmic approach). In this way large steps towards finding a solution could be made when the ship planning process for a given port begins, with each step being reduced in size as the cargo-space fills. This process is illustrated in Figure 8-8, where large block spaces at the start of the search imply many possible moves (container placements) to new states. As the spaces fill, the number of possible moves reduces.

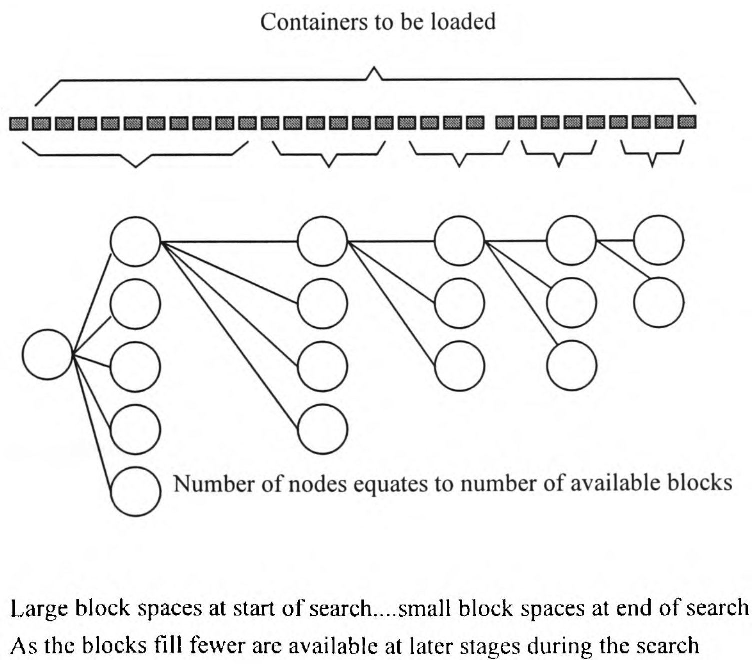


Figure 8-8 Illustration of changing increment applied to planning

Evaluation of each state during the search process will be based upon the criteria described in Section 7.4. The above process produces many different stowage configurations for a single port. In order to evaluate the effectiveness of the solutions, consideration must be given to stowage configurations at subsequent ports. A new pool of feasible solutions will be generated at each port-of-call for each of the arrival solutions (illustrated in Figure 8-9), (the number being dependant upon how distant the port under consideration is from the container-ships actual terminal, available time and processing resources).

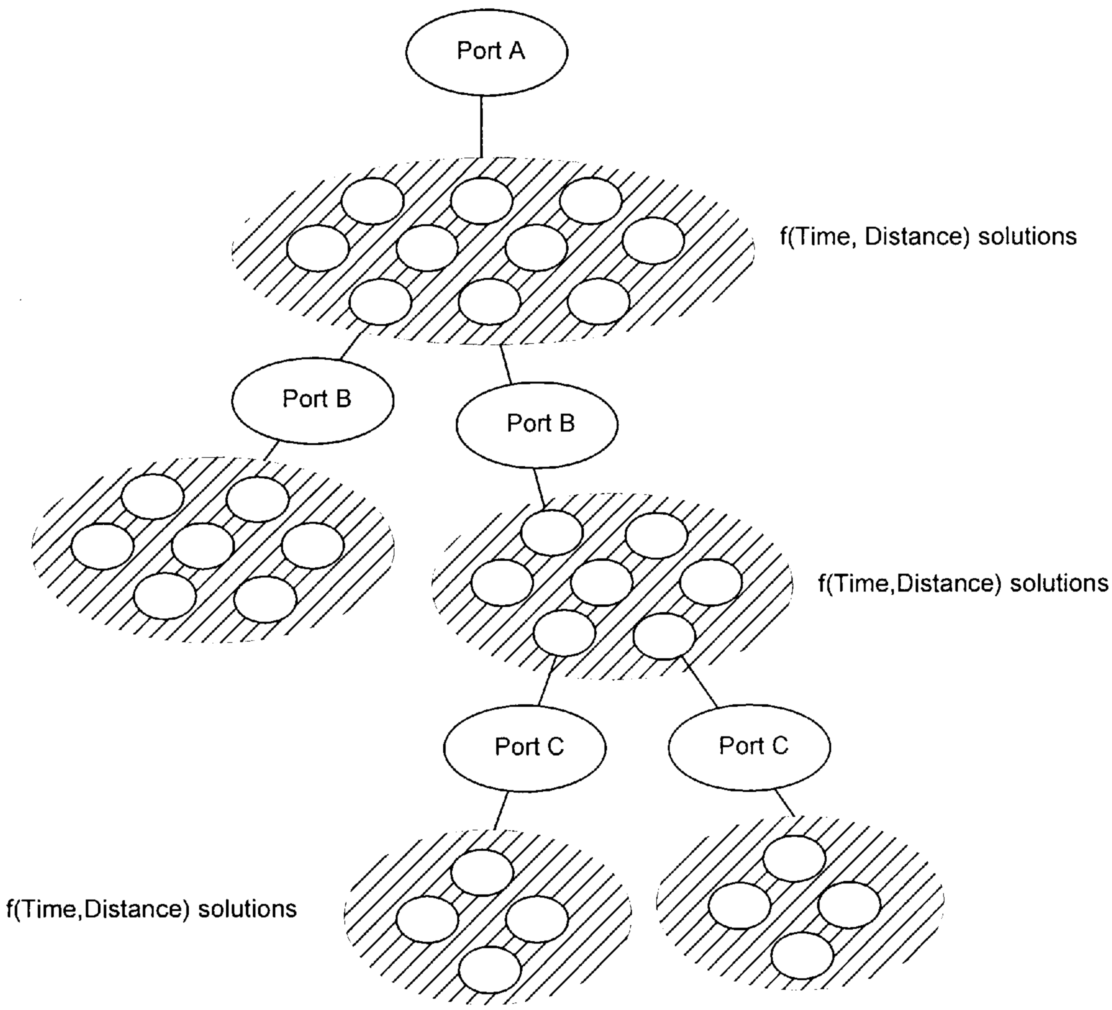
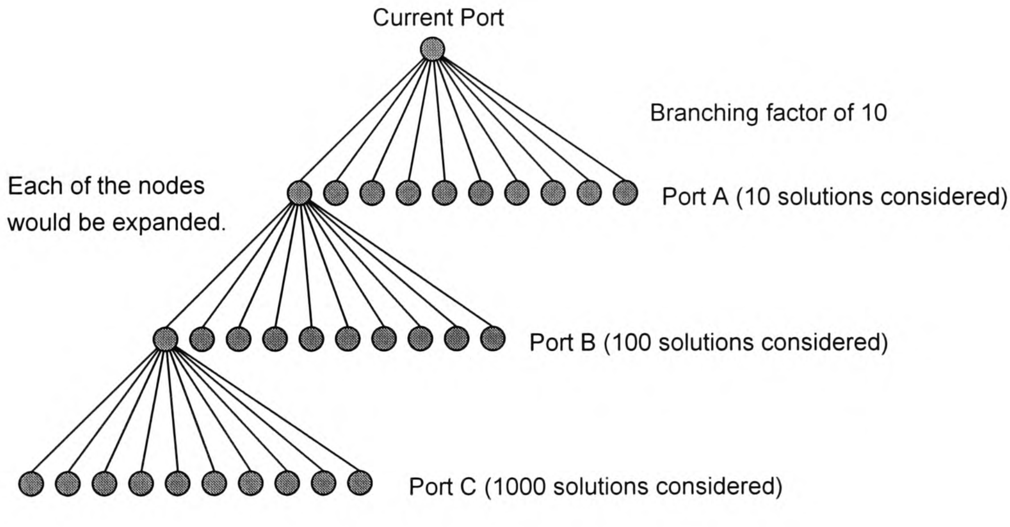


Figure 8-9 State-space showing fewer generated solutions at future ports

To explain why it is important to generate fewer solutions at further ports it is necessary to consider the following. If each of the solutions generated at the previous terminal are processed, then a linear multiplication of stored solutions would result (*e.g.* 10 of the best solutions generated at terminal A are passed for processing to terminal B, each of these solutions are used as a starting point generating 10 new solutions for each of 10 arrival solutions resulting in 100 solutions for the next terminal. Should this process continue, then 1000 solutions would be passed on from terminal C to terminal D, 10000 from D to E and so on). Attempting to explore this vast state-space (illustrated in Figure 8-10) would be prohibitive.



Therefore, given the heavy reliance placed upon statistical information for planning at more distant terminals it is reasonable to reduce the number of generated solutions passed on for processing to each subsequent terminal in the considered journey. Hence the situation shown in Figure 8-9 (and illustrated again in Figure 8-11). The exact number of solutions considered at each branch in the tree would depend upon available time and resources, accuracy of statistical information and experimentation.

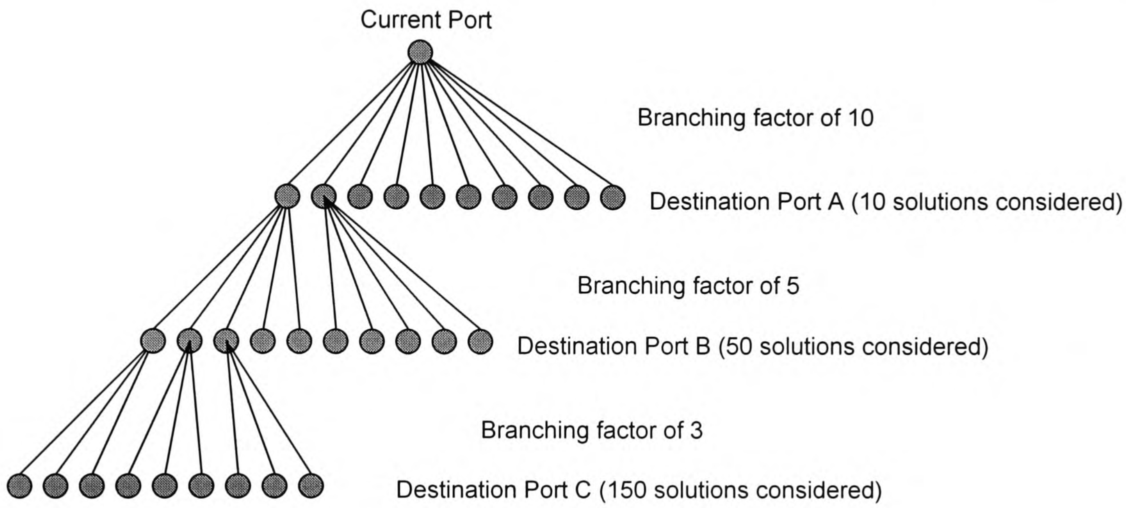


Figure 8-11 Branching factor reduced with distance

It is important to remember that each of the solutions passed on for consideration at the next container-terminal are starting points from which a large number of solutions will be generated, from which the best *n* solutions are passed on. Each solution within the pool will then be ranked according to the cost, in real terms, associated with:

- crane utilisation;
- hatch-lid movement;
- and number of restows generated.

Ranking will be determined by simulating the loading and unloading process at the each destination terminal, summing the overall cost. The solution selected as best at the current port-of-call will be the one that ranked the highest when considering future ports, not the best when only considering the next terminal.

Constraints upon what cargo types (for example, some cargo types taint their surroundings and require specially treated areas for stowage) can be stored in each block will be used to prune branches from the search tree. Figure 8-12 illustrates this process, where each combination of cargo placement into a cut-out from a longitudinal cross-section of a container-ship is shown, in this case four hatches. Each of the new generated states is then examined to ensure that no constraint has been broken and, in this case, two states are removed from the tree since tainting cargo has been placed into these two cargo-spaces.

Bending moments and trim will be calculated for each final solution for a given port, starting with the most promising and continuing to the next promising until all constraints are passed and a pre-determined number of solutions are set aside for processing at the next destination. The number of solutions set aside for processing at the next port-of-call will depend upon available computer architecture and processing time. In this way the long-term consequences of initially promising solutions will be determined and a given number of these strategic solutions will be set aside for latitudinal processing stage described in the next section. By not pursuing all solutions through consequences at successive ports, initially unpromising solutions which later would be seen as cost effective would be rejected.

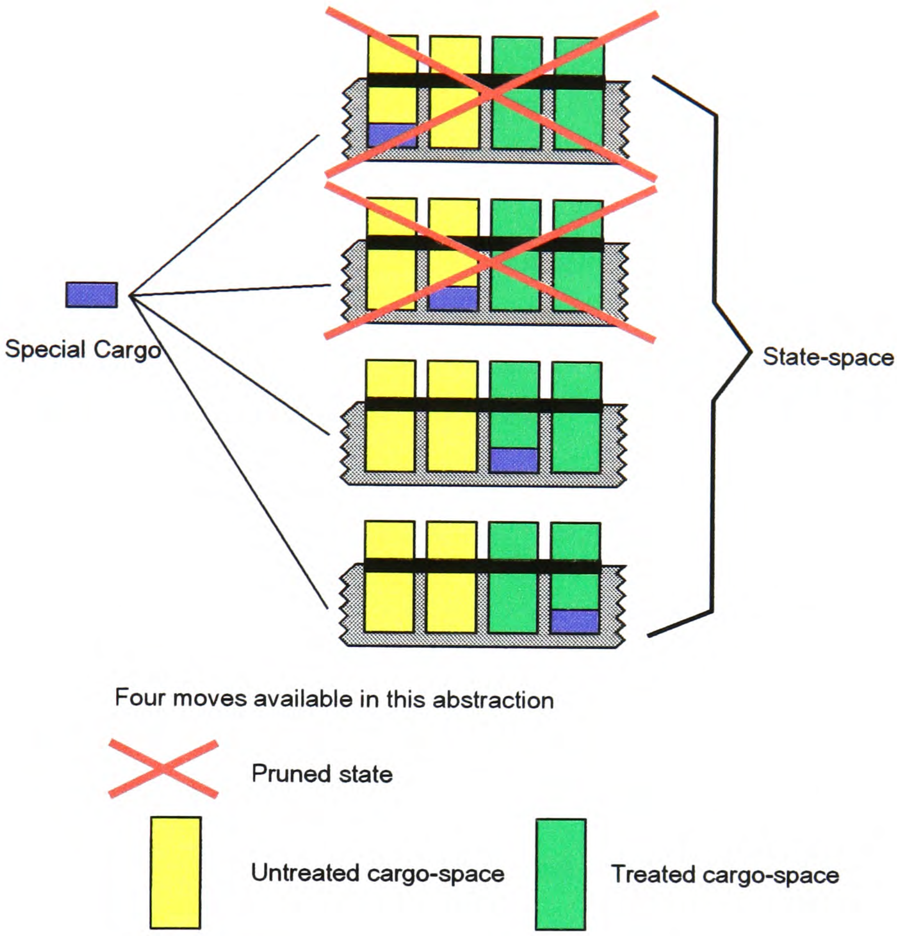


Figure 8-12 State-space branch pruning

Therefore, an optimum solution to the overall stowage problem could not be guaranteed. However, the size of the state-space associated with processing all possible paths makes this prohibitive. In addition, the high degree of reliance upon statistical information reduces the significance of solutions generated at future ports making it less productive to explore all possible combinations of possible paths.

8.2.3.2 Applying search to the latitudinal abstraction

The search process described in Section 8.2.2.3 will produce a number of valid solutions that will each be processed latitudinally, according to the latitudinal abstraction (described in Section 8.2.2.4). Each of the containers stowed in a given hatch will be allocated to a particular block using a Branch and Bound search (described in Section 4.2.3) where evaluation of each completed solution is based upon the following criteria:

- that cargo-space usage should be maximised;
- that hatch-lid movement should be minimised;
- that over-stowage should be avoided.

During the process of generating solutions, invalid solutions will be pruned from the search-tree (illustrated in Figure 8-12). Different valid solutions will be generated until all are exhausted or available time has expired. Within each hatch, search will be applied to generate stowage configurations that meet the above criteria and minimise heeling on a local level (to this end, weight is distributed athwartships between blocks as evenly as possible). Constraints upon the type of container and cargo that can be placed in a given block will actually assist in reducing the state-space of the problem. The most promising blocked outline-plan (similar to that shown in Figure 8-7) will be checked for intact stress and stability. It is assumed that in practice, ballast will be used, if necessary, to bring the stowage pattern to within tolerable levels. However, it is accepted by this author that the use of ballast should be kept to a minimum (unlike the view expressed in Botter [36]) and this should be reflected in the evaluation of a stowage pattern. This process will be repeated for a given number of the best generated stowage patterns with the single solution that offers the best overall condition being selected for the tactical planning phase described in the following sections.

8.3 Automating the tactical planning phase

Section 8.2 described how the strategic planning phase of the solution to the deep-sea container-ship stowage problem would be implemented. The overall solution also requires the implementation of the *tactical planning* phase, which is dealt with in this section. In the tactical planning phase (explained in detail in Section 3.5) the human planner takes the Outline Plan, where classes of containers are allocated to stowage locations and generates specific Bay Plans (described in Section 3.3.2.3) where the individual locations of specific (no longer just a class) containers are shown. The author uses here a heuristic based approach that constitutes the first phase of making individual placements which is based on *packing theory* [37] (the placement of different sized objects into a three dimensional space, often referred to in literature as Back-Packing). The objective here is to generate a starting point for optimisation. This heuristic-based approach provides a solution which satisfies stack weight and height constraints whilst minimising overstows and void stowage locations (brought about by out-of-gauge containers partially extending into adjacent slots). This solution is not optimal, but is necessary for the next phase of tactical planning where the quality of the 'packed' cargo-space is improved by progressive improvement.

8.3.1 Heuristic based placement of containers within blocks

The tactical planning phase uses the best solution found in the strategic planning phase. However, if time allows, it may also consider other possible solutions from the strategic planning phase. Essentially, the strategic planning phase (described in Section 8.2) provides a series of alternative partial solutions to the container-ship stowage problem. Each partial solution has all containers to be loaded (expressed in terms of their general classes) assigned to blocks . Given one of these individual blocks (an example of which is shown in Figure 8-13) and the containers allocated to it, tactical planning allocates specific containers to individual slots.

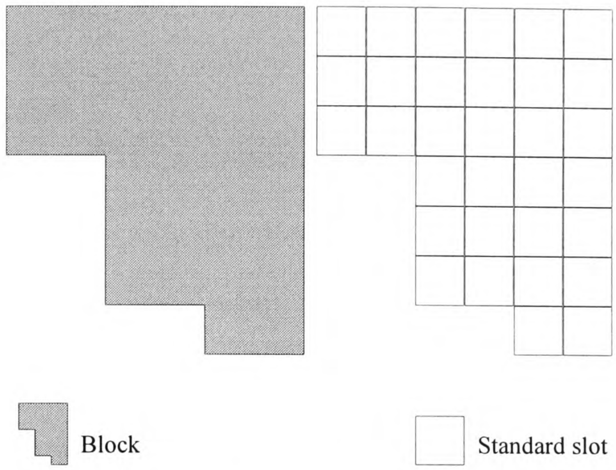


Figure 8-13 Cellular structure of a cargo block

Three-dimensional packing [37] is a common research problem. Typically, solutions are found through the application of problem specific heuristics. Much of the published work in this area has been concerned with the packing of shipping containers, often referred to as the *container stuffing problem* (illustrated in Figure 8-14).[55,56,57] These approaches are predominantly based upon a series of ad hoc heuristic rules derived by common sense; it is therefore unsurprising that no single approach can be said to be superior to others. [54]

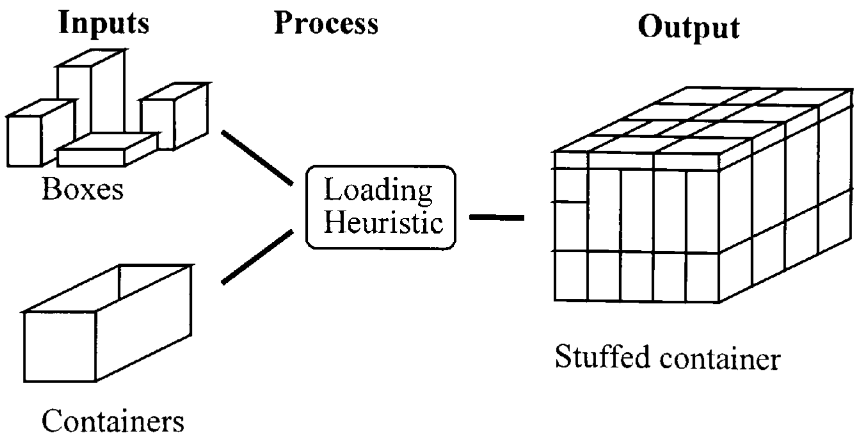


Figure 8-14 Example of 3D packing

The following section describes a loading heuristic, based upon three dimensional packing theory, that allocates stowage slots to blocked containers.

8.3.2 The cargo-space packing heuristic

This section proposes a new loading algorithm based upon container-ship loading heuristics specific to this project. It will be shown here that the heuristics used are derived through a 'common-sense' approach to the problem. Specifically, an analysis of the stowage objectives given in Section 3.2 will be used in designing these heuristics. The exact heuristics used to load a block depend upon where the block is located (above or below deck; starboard, central or port) and any special requirements associated with the blocked containers (such as, requiring access to external power sources). The primary objective of this phase in the planning process is to obtain a solution that is a starting point. That solution can then be altered during an optimisation phase. The blocks which are under hatch-lids will generally have restrictions upon the length and width of container that can be placed there. However, it should be noted that containers are often nothing more than frames and may have cargo protruding from the sides, thereby exceeding notional dimensions (refer to Section 3.3.2.2 for information about how over-height and width containers are marked on plans). This type of container is often accommodated in under hatch-lid blocks, but this results in a container protruding into adjacent blocks. Therefore a loading heuristic must minimise the number of slots affected (voided) by this type of container. Over hatch-lid blocks have no restrictions upon the dimension of container that can be accommodated. All over-width containers will be placed so that they do not protrude into adjacent blocks (since, by doing so, they would count against allocated TEU capacity for the affected block). Hazardous containers are more likely to be stowed in over hatch-lid blocks, typically to the extreme port or starboard (particularly cargo that may put the ship at risk, the solution then being to push the container over-board).

The rules of thumb (or heuristics) used to develop a loading heuristic are that:

- containers should not be stacked on top of other containers which will be discharged first;
- heavier containers should be stowed lower in the block than lighter containers;
- an over-width container should be placed in such a way as it is the last in a full stack, thus reducing the affected slots to those immediately adjacent to the container (see
- Figure 8-15);
- containers of the same length should be stacked together.

One way of encoding these heuristics is as described below.

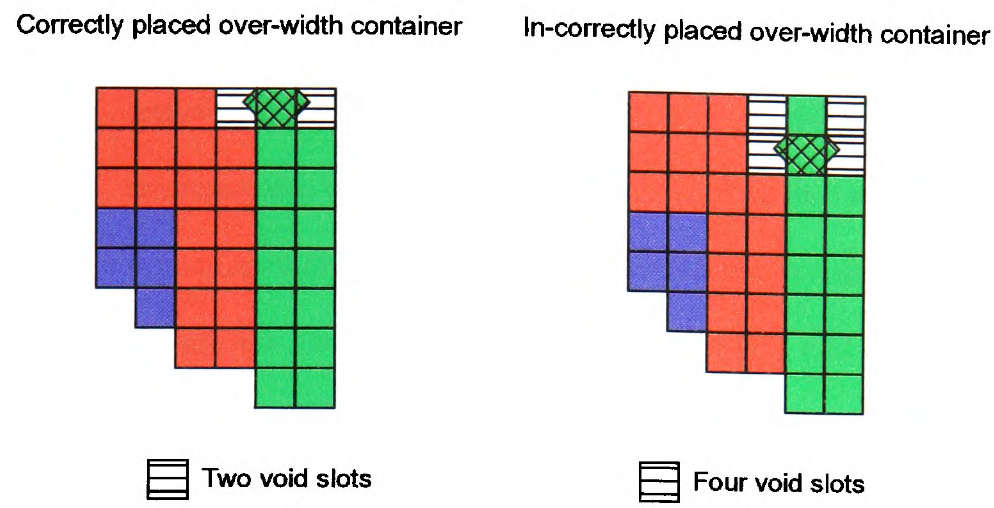


Figure 8-15 Diagram showing alternative voided slots

Before allocation of containers to slots begins, the author proposes that the containers to be loaded be sorted according to standard size (largest first), destination (furthest first) and weight (heaviest first). Then, dependant upon which block is being filled (port, starboard or centre), containers are sequenced into the block using a heuristic of the following form (typical results of which are illustrated in Figure 8-16) for each block:

1. Choose a stacking order appropriate to the block under consideration:

   port side: stack upwards, stern to bow, from right to left.

   central: stack upwards, stern to bow, centre outwards.

   starboard: stack upwards, stern to bow, left to right.

2. Find the first stack that is not full.
3. Find the next container in the load-list - this becomes the container under consideration.
4. If there are still standard containers to be loaded then find any stack that is not full whose top-of-stack container is destined for a discharge port the

same as or further than the container under consideration. Place the container under consideration in the stack. Attempt to fill this stack with any remaining containers of this class taken from the load-list.

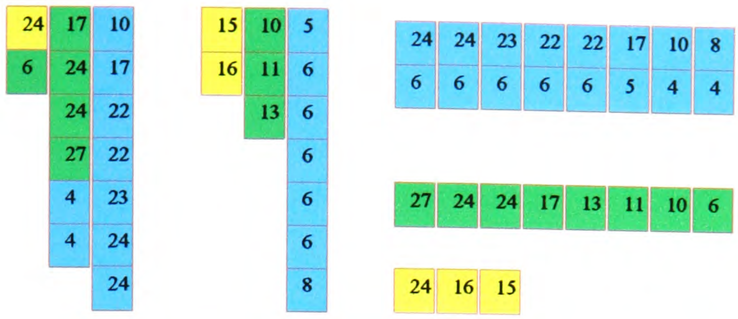
5. If there are still standard containers to be loaded then goto step 3.
6. If there are out-of-gauge (OOG.) containers then find a full stack where the top-of-stack container has the same destination, or further, as the OOG. container and attempt to swap them; if they can be legally swapped (taking into consideration adjacent stacks) then do so and place the removed container back into the load-list.
7. If there are still containers in the OOG. list then goto 6
8. If there are containers in the load-list then goto 3

In this way all containers will be allocated a relative location (one that is relative to other containers, not necessarily conforming to cell-guides) in the cargo-space such that:

- overstows are minimised (since containers with the furthest to travel are loaded first onto stacks that themselves do not have any containers there that will generate restows);
- void spaces are minimised (since over-width containers will always tend towards top of stacks where there effects are kept to a minimal);
- heavier containers are stowed lower than lighter ones (since the load-list is ordered in such a way that heavier containers are loaded first);
- stacks are, generally, made up of containers of the same class.

The exact manner of loading can be varied, requiring changes to the above algorithm, in order to define different heuristics. The following illustrations show the state of a bay loaded using three different heuristics. The same containers, in this case all the containers to be loaded are twenty foot long and correspond to cell guides (*i.e.* are not out-of-gauge), are placed into the same cargo-space. The load-list holds containers with three different destinations (where colour denotes the destination), with each group of destinations being sorted so that heavier containers (the number within the square representing a container shows the weight in tons) are at the front.

Load heuristic number one (shown in Figure 8-16) simply fills the stacks upwards, aft to fore, from the centre of the ship outwards. No restows are generated here although heavier containers tend to be placed over lighter ones.



Fill inner stacks first, aft to fore.

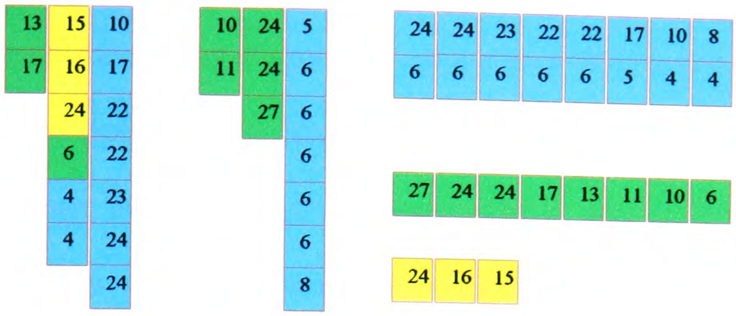
Figure 8-16 Stacking heuristic-variant one

Load heuristic number two (shown in Figure 8-17) loads the central stacks, tier by tier, aft to fore, instead of stack by stack as in heuristic one. (Changes in the algorithm are required for this.) Again, no restows are generated but heavier containers tend towards the bottom of stacks here.

Fill centre stacks, tier by tier, upwards, moving outwards when inner stacks are full.

Figure 8-17 Stacking heuristic-variant two

Load heuristic three (shown in Figure 8-18) sees a new stack being selected when a new container destination is encountered. (Changes to the algorithm are required for this.) This method would tend to keep containers with the same destination together.



Stack upwards, from stern to bow, centre outwards.

Begin a new stack with every new destination.

Fill up incomplete stacks using the same procedure as above.

Figure 8-18 Stacking heuristic-variant three

All three heuristics generate different starting points for the optimisation process described in Section 8.3.3. Each heuristic generates a stowage configuration in a very short time usually allowing all to be tried out with the best resultant stowage configuration being used as a starting point for the optimisation process. A number of factors, such as the composition of the load-list or the cargo-space already having cargo stowed there, may influence the relative success of each heuristic. Any of the example heuristics given in this section are sufficient to demonstrate the applicability of this approach and generate a starting point for the optimisation process. The choice of heuristic used to place containers into the cargo-space could, in itself, produce a good stowage solution. However, this solution is unlikely to be optimal.

The solution found at this stage, therefore, is used as a starting point for an optimisation process which rearranges the containers in the cargo-space. This optimisation process is explained in the next section.

8.3.3 Optimisation of the heuristic generated stowage configuration

The loading heuristic generated stowage configuration will require alteration since it is unlikely to be optimal, may have illegal relationships (hazardous segregation) between containers and may require containers to be moved to specific locations (such as containers requiring external power sources). Since human planners conceptually swap containers around until all constraints are satisfied and a near optimal solution is generated, a search methodology that models this approach would offer the most promise. Tabu search [59,60,61,62] is a well known optimisation process that adopts a similar approach to that of the human planner. This method is outlined in Section 8.3.3.1 below prior to describing how this optimisation method can be applied to the blocked stowage configurations generated by the hybrid approach described in Section 8.2.3 and Section 8.3.

8.3.3.1 Introduction to Tabu search

Tabu search [59,60,61,62] can be used to guide any process that employs a set of moves for transforming one solution (or state) into another and that provides an evaluation function for measuring the attractiveness of these moves. [58] Tabu search is conceptually similar to Hill Climbing (introduced in Section 4.2.2) where the search process continues until a destination state or non-improving state is discovered. The main difference between these methods is that in Tabu Search a given number of additional non-improving moves will be made in the hope that a destination state better than the best previously encountered will be discovered. The use of non-improving moves is illustrated in Figure 8-19, in which the graph shows a 'cost' of a solution after each change (move). Lower costs represent better solutions. Hill

Climbing would stop soon after the 'local best' indicated, but Tabu Search might continue over the peak of worsening solutions to reach the 'global best'.

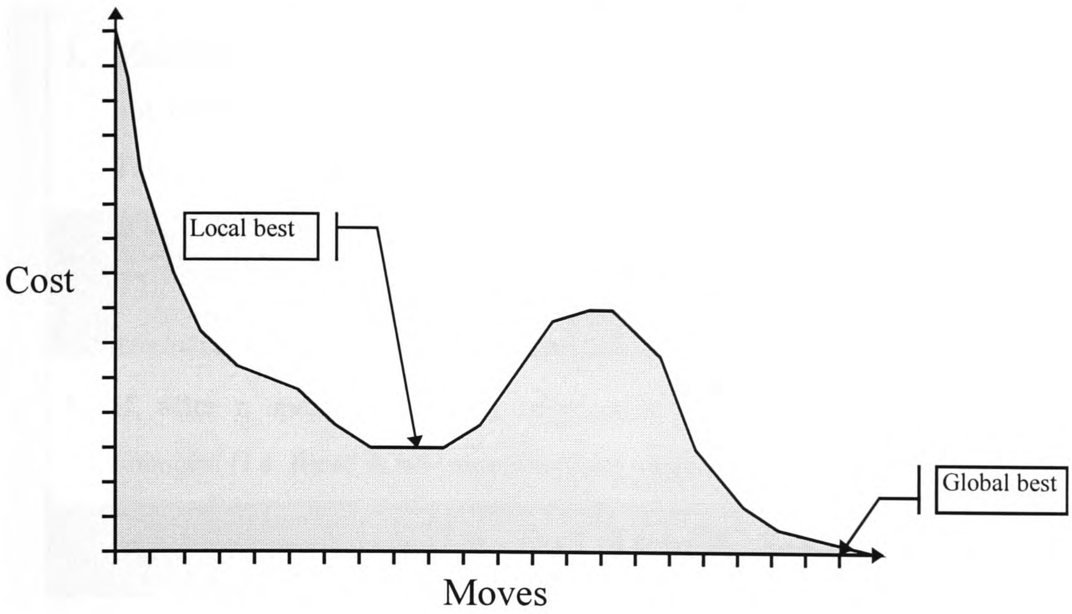


Figure 8-19 Graph showing cost of solution after each move

The strength of Tabu search (the algorithm for which is described in Table 8-1) lies within its use of short-term memory to store previous moves. This short-term memory is used to prevent the reversal, or repetition, of moves made recently. These Tabu restrictions permit the search process to go beyond points of local optimality by making high quality, non-improving moves, hopefully towards a global optimum solution. Unless recent moves are rendered inadmissible (or 'Tabu'), the search process could make a 'best' (non-improving) move away from a local optimum and then fall back into the 'local best' when later moves are made. In cases where none of the available moves are admissible, then a least worse 'inadmissible' move (saved to handle such as possibility) is chosen instead.

1. Begin with a starting solution and save this solution as the best so far.
2. Create a candidate list of possible moves from the best solution so far.
3. Make the best admissible move (one that is not Tabu or is a Tabu move but improves upon the solution generated by the same move held in the Tabu list).
4. If the new solution is better than the old best solution then replace the old best solution with the new solution.
5. If, after a specified number of moves, the best solution has not been changed (*i.e.* there is no improvement), terminate the search.
6. If the search has not terminated, update the Tabu list and goto 2.

Table 8-1 Basic Tabu search algorithm

A so-called *aspiration* level [51] can be associated with each move contained within the Tabu list. This is to say that the evaluation score of a solution generated by a particular move (*e.g.* container swap) is stored with the move details within the Tabu list creating a pair, being the move and the associated score for the state of the problem that move generated. An otherwise Tabu move (one that has been made recently) is selected if the evaluation score for the new state is better than the evaluation score associated with that same move stored in the Tabu list. (*e.g.* Elements x and y were swapped at move number 8 generating a state with an evaluation score of 10, later in the search process no improving move that is not Tabu can be found so the Tabu list is checked. The move 'Swap x and y' is in the Tabu list but since the state has moved on, repeating this same move now would generate an evaluation score of 8, 2 better than the score earlier associated with this particular swap. Therefore, 'swap x and y' is selected as the new move.) The following section describes how Tabu search was used to optimise the block stowage of containers within a heuristically filled cargo-space.

8.3.3.2 Using search to optimise heuristically generated stowage plans

Tabu search requires a valid solution to begin with, hence the need for first generating a stowage solution by the planning heuristic detailed in Section 8.3.2. Given a heuristically filled cargo-space (such as the starboard, under-deck twin bay Figure 8-20), the task of the optimisation process is to rearrange the containers until such time as no further improvement is expected.

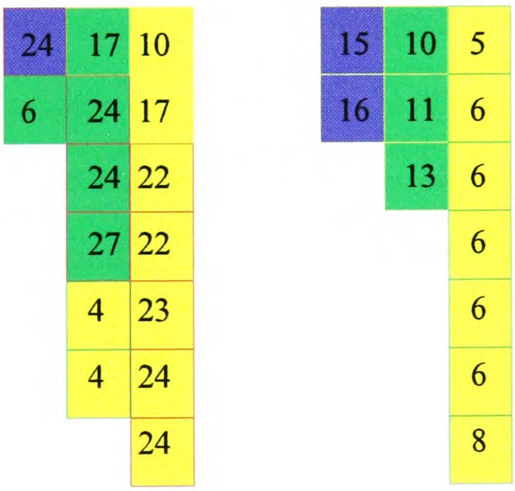


Figure 8-20 Example of a heuristically filled cargo-space

Tabu search is designed to produce solutions for large combinatorial problems [51], the key to the process being the determination of a set (or *neighbourhood*) of moves from the current state that are admissible. The definition of what constitutes a neighbourhood of admissible moves facilitated the process of problem decomposition undertaken by the author. The neighbourhood, in the context of the stowage-planning problem, is the set of permissible moves of containers within a single block of cargo-space (illustrated in Figure 8-20). The state-space associated with a single block is small enough that a number of optimisation algorithms will, in theory, be able to find an optimal solution. However, the Tabu search algorithm was selected for implementation and experimentation since it closely models the conceptual processes (where a stowage pattern is progressively altered by moving containers around) performed by the human planner.

8.3.3.3 Evaluating the stowage pattern of a single cargo-space

The attractiveness of a valid stowage solution can be measured in a number of ways, for example:

- the number of overstows associated with a stowage configuration gives a good indication of its attractiveness;
- the number of stacks with containers of mixed length can be used to further evaluate the attractiveness of a stowage configuration, this being considered undesirable by planners;
- counting the number of lighter containers stowed below heavier containers gives an indication of adverse stability;
- measuring the distribution of weight athwartships also gives an indication of stability and can easily be evaluated.

The next section describes a phased approach to stowage optimisation using Tabu search.

8.3.3.4 Phased cargo-space stowage optimisation

The evaluation criteria described in Section 8.3.3.3 could be incorporated into one evaluation function, with different weightings assigned to each criteria. Alternatively, the criteria could be used to further reduce the neighbourhood of permissible moves by phasing the optimisation process (described in Table 8-2).

Phase one	Optimise the cargo-space by moving containers around until the number of overstows has been reduced to a minimum. The neighbourhood at this stage in the search process is the set of all moves of all containers within the cargo-space.
Phase two	Optimise the cargo-space by moving containers around until as many stacks as possible have the same length of container stowed there. (Where the neighbourhood for a particular swap is restricted to containers with the same destination and, where appropriate, special type.)
Phase three	Within each stack of containers arrange containers with the same destination so that heavier containers are stowed below lighter ones. The neighbourhood here being limited to same destination containers stowed in the same stack.
Phase four	Optimise the cargo-space by moving containers around until the weight distribution of containers facilitates attaining intact stability. The neighbourhood at this stage in the search process is restricted to moves of containers with the same destination.

Table 8-2 Phased cargo-space optimisation

A 'move' is either movement of a container to another vacant location or a swapping of a single container with a number of containers that are equivalent in volume (*e.g.* a 40' long container could be swapped with two 20' containers). Since a typical cargo-space block will hold approximately 12-60 TEU's, finding an optimum solution given the above criteria is a relatively simple combinatorial task. Experimentation with the Tabu search algorithm generated optimum solutions, when considering 20' under-deck bays, in few iterations (experimentation with small problems needing as few as 15 iterations). This form of optimisation lends itself well to the shipping operator that is still accepting containers for transport whilst the loading process is in progress and leaves the planning of bay-plans to the last moment - each bay is placed in sequence as the precise container details become available.

Experimentation with Tabu search applied to the optimisation of the stowage of containers within pre-assigned blocks resulted in the rapid generation of optimal stowage configurations (where over-stowage was minimised and weight distribution maximised).

8.4 Intact stability and ballast

Part of the function of the longitudinal block stowage algorithm described in Section 8.2.2.3 was to distribute the weight of cargo along the length of a vessel so that an acceptable trim (explained in Section 2.3.5) be found and bending moments (explained in Section 2.3.6.2) be minimised. Similarly, part of the function of the latitudinal blocking algorithm was to minimise heeling moments (explained in Section 2.3.5.2). Although the loading heuristic described in Section 8.3.1 has little effect upon intact stability it does act as an input to the optimisation algorithm described in Section 8.3.3 where part of the optimisation process is dedicated to distributing containers with intact stability as a primary evaluative factor. Given that each phase of the load-planning process described above considers intact stress and stability it is the writers belief that any further adjustments required to achieve intact stress and stability will be minimal and could reasonably be effected by alteration in ballasting. How a rule-based expert system can be used to perform ballasting for an oil-tanker has been introduced in 5.6.1. Little difficulty is envisaged in implementing a similar system specific to container-ships.

8.5 Processing hazardous cargo segregation requirements

All four levels of hazardous segregation (described in Section 2.1.3.3) are incorporated into the proposed system. The longitudinal blocking process described in Section 8.2.2.3 accommodates class 3 and 4 requirements (respectively, that containers being separated by a complete compartment or hold from and separated longitudinally by an intervening complete compartment from). Similarly, the optimisation process described in Section 8.3.3 allows class 1 & 2 requirements to be accommodated (respectively, that containers be stowed away from and separated from). In both cases, after a hazardous container placement is made the relative locations of other hazardous containers must be checked against segregation requirements. This can easily be accomplished by storing the locations of hazardous containers in *lookup-tables* in addition to the primary data-structure representing the cargo-space. The validity of a new solution, in the context of hazardous segregation requirements, can then be readily determined with a minimum of processing by consulting the information contained in the look-up table. Organising the look-up tables into trees will make the process of checking hazardous cargo segregation constraints more efficient.

8.6 Conclusion

The proposed stowage-planning model mirrors how the paper documents are used by human planners during the planning process. Other authors, when dealing with stowage planning, have attempted to rely upon powerful computers to develop stowage solutions, still requiring that important problem features are left out in an attempt to reduce the problem to a scale that is solvable. These attempts to reduce the problem in size and complexity until it fits the chosen hardware and software platform have resulted in failed implementations and a consequent drying up of research into automating stowage planning (at least from the operator's point of view). Each of the documents used by the human planner deals with a different conceptual level of the problem, from the most general (General Arrangement) to the most specific (Bay Plan). Modelling the processes undertaken at each stage in the planning process, identified by the particular document used, allows the automation of the whole process without elements of the problem being discarded.

The hybrid approach, described in this chapter, that incorporates a wide variety of search algorithms to a decomposed model of the container-ship stowage problem that corresponds to the procedures and documents used by the human planner appears to offer the most promising results. Experimentation with prototype search procedures applied to each of the sub-processes described in this Chapter demonstrated the feasibility of this approach.

The next chapter outlines the experience gained from the project and the conclusions that can be drawn from applying search to this type of 'real world' problem.

9 EVALUATION, CONCLUSION AND FUTURE WORK

9.1 Overview

The deep-sea container loading problem is combinatorially explosive with the number of possible stowage configurations for a medium-sized container-ship of 2000 TEU being vast (approximately 3.3 times ten to the 5735th power [49]) and can, therefore, be described as being NP-Hard. [36] This is to say that an optimal solution can not be found for commercial sized ships in a reasonable processing time using commonly available computer software and hardware. However, a hybrid approach incorporating different AI techniques at each level of the decomposed model of the stowage planning problem appears to offer good, if not optimal, results.

The problem is complex due to the large number of variables, such as vessel intact-stability, hazardous cargo segregation and the need to separate cargo intelligently to permit efficient manipulation at ports, that require consideration. Added to the knowledge intensive aspect of the problem is the multitude of theoretically plausible solutions, or stowage plans, that are available to the planner. Given the combinatorial complexity of the problem, the search for a solution that approximates to the optimal made this problem a difficult, but worthwhile, one to research.

This stowage planning task exhibits two classic search elements: constraint handling and the use of heuristics. These heuristics and constraints are treated as independent components within the stowage planning system since these factors vary from operator to operator. The envisaged system for solving the stowage problem is knowledge intensive, requiring a variety of different inputs. The system will require forecast information that will be provided by the shipping company. Therefore, some suitable interface and appropriate protocol would have to be developed in order that this information can be made available to the system. The system will require information about the containers that are to be loaded whether the information be actual or forecast. Issues concerning this interface were considered to be beyond the scope of this project.

This chapter summarises the results of the design and testing of a computerised planning system that solves the deep-sea container-ship planning problem. In particular, it highlights how general Artificial Intelligence problem solving algorithms can be employed for this purpose.

9.2 Related work

The analysis given in Chapter 5 of work relating to computerised cargo-ship planning applications demonstrates the inadequacies of those attempts. Important common denominators can be identified within these stowage planning applications. These similarities highlight the inherent difficulties associated with each approach to solving the container-ship stowage planning problem and are, again, summarised here:

- the combinatorial complexity involved considering each conceivable stowage configuration is highlighted;

- salient domain features and constraints are identified;
- the representation of the problem is simplified by only considering the placement of standard containers into standard cellular container-ship stowage locations (thus, disregarding out-of-gauge containers);
- the representation of the problem is simplified by limiting consideration of special containers to making provision for external power sources (thus, disregarding a wide variety of cargo-types requiring, for example, specially treated cargo-spaces);
- the representation of the problem is simplified by not making provision for the segregation of hazardous containers;
- no true analysis of how and why human planners prepare stowage plans, with reference to existing systems and documents there in, resulted in the above simplifications.

9.3 Expert system development

Any computerised implementation that attempts to emulate a human decision making process falls under the label of an Expert System. At the heart of Expert System development lies the acquisition and representation of models of domain expertise, processes and structures. To set up hypotheses, to choose one that is the most favourable and to modify it after verification is sophisticated cognitive work and is not suitable for formulation using conventional programming methods. For the problem of this project, the application of advanced search techniques would appear to offer the best possibilities. Other researchers investigating the stowage planning task have failed to fully conceptualise the problem, instead choosing to simplify it (such as Botter & Brinati [36]) or, where the problem is fully understood, do not have sufficient knowledge of how AI can be exploited to produce workable implementations. [1, 13, 43, 45, 49, 52]

Given the complexity of the container-loading problem, and the close inter-dependencies of the data representations and the effectiveness of solutions, a substantial part of this thesis is dedicated to a comprehensive knowledge elicitation exercise that was performed in parallel with other experimental work. During this exercise, criteria for evaluating stowage solutions was identified and incorporated into a suitable evaluation function. The results of this knowledge elicitation and detailed problem analysis process, as recorded in this thesis, constitute a cleaner and more complete statement of the problem and system implications than appears elsewhere in the literature. The author strongly believes that this represents a substantial contribution to the field, hence the considerable space allocated to this work in the thesis.

This thesis introduced a number of related issues which affect the stowage of cargo on container ships. It can be seen that the problem is of a non-trivial nature. It was determined that an optimum stowage pattern for an outbound container-ship could be determined through the application of search techniques. However, given that a medium sized container-ship may have to load upwards of six hundred containers into, perhaps, eight hundred possible locations, it can be seen that this is a massive computational problem where an optimum solution, based upon the entire state-space, is unlikely to be found in a realistic time frame.

General search theory was applied to representative abstractions of the stowage planning problem with important lessons being learnt that highlighted the inadequacies of other authors' work and stressed the importance of drawing upon existing, human planner, based systems.

The approach taken designing the proposed stowage planning model mirrors the development of the paper documents used by human planners during the planning process. Other authors, when dealing with stowage planning, have attempted to rely upon powerful computers to develop stowage solutions with the result that important problem features are left out in an attempt to reduce the problem to a scale that is solvable. These attempts to reduce the problem in size and complexity until it fits the chosen hardware and software platform has resulted in failed implementations and a consequent drying up of research into automating stowage planning (at least from the operator's point of view). Each of the documents used by the human planner deals with a different conceptual level of the problem, from the most general (General Arrangement) to the most specific (Bay Plan). Modelling the processes undertaken at each stage in the planning process, identified by the particular document used, allows the automation of the whole process without elements of the problem being discarded.

The hybrid approach, described in Chapter 8, incorporates a variety of search algorithms to a decomposed model of the container-ship stowage problem. The hybrid approach corresponds to the procedures and documents used by the human planner and appears to offer the most promising results when compared with the approaches taken by other authors. Experimentation with prototype search procedures applied to each of the sub-processes described in this Chapter demonstrated the feasibility of this approach.

9.4 Assessment of the proposed decomposition

In container-ship management, one is confronted with the problem of how to assign to each container a reasonable position in the container ship, so that the efficiencies of later container handling operations are increased. At the same time, some necessary conditions must be satisfied (*e.g.* space limitations within the container ship, compatibility requirements of the containers that are stowed together, *etc.*). The feasibility of the automated planning methodology proposed by the author is demonstrated by the following, representative, worked example.

9.4.1 Problem size and computational complexity

The container stowage problem, as described in this thesis, is a combinatorial problem, the size of which depends upon ship capacity (given by the number of TEU units) and the container supply and demand at each port of the route. Even for the smallest cases, the container stowage problem, from the point of view of combinatorial optimisation when considering the stowage of individual containers across a number of ports, is a large-scale problem.

Determining the optimum allocation of specific containers to slots over even a few ports is computationally explosive and is not solvable in a realistic length of time. In solving this problem, the author has sought to produce good, but not necessarily optimal, solutions.

9.4.2 Decomposition of the complete problem

In order that the computational difficulties associated with producing a solution for the stowage problem can be over-come, the author has proposed a decomposition of the stowage planning process (explained in Chapter 8), namely:

1. A strategic planning process involving the assignment of generalised containers to a blocked cargo-space;
2. A tactical planning process involving the assignment of specific containers to specific slots within their assigned blocked cargo-space.

The solution of the strategic planning phase gives a picture of the generalised cargo stowage distribution at the end of unloading and loading processes at each port on the route. This approach models the way human planners perform the planning process and reduces the combinatorial size of the problem whilst retaining the inherent characteristics of the problem. Blocking the cargo-space of the container-ship enables the number of decisions available at any stage of planning process to be reduced from, perhaps, thousands of possibilities to within a hundred.

The second, tactical planning, phase determines the exact slot occupied by each container at current port-of-call. Notice that, unlike when the whole ship is considered, the combinatorial nature of the problem is now reduced to allocating specific containers only within a small part of the container-ship. Again, this approach models the way human planners approach the problem.

9.4.3 Description of the theoretical example

Since container-ships and the routes they service vary considerably, it is important that the underlying planning process be understood and modelled. In practice, further knowledge specific to a particular trade route and ship operator could then be added. The hypothetical container-ship described below and illustrated in Figure 9-1 (in an Outline Plan) is sufficiently large, and has sufficient cargo-space variability, for illustrating the underlying planning methodology and supporting data-structures.

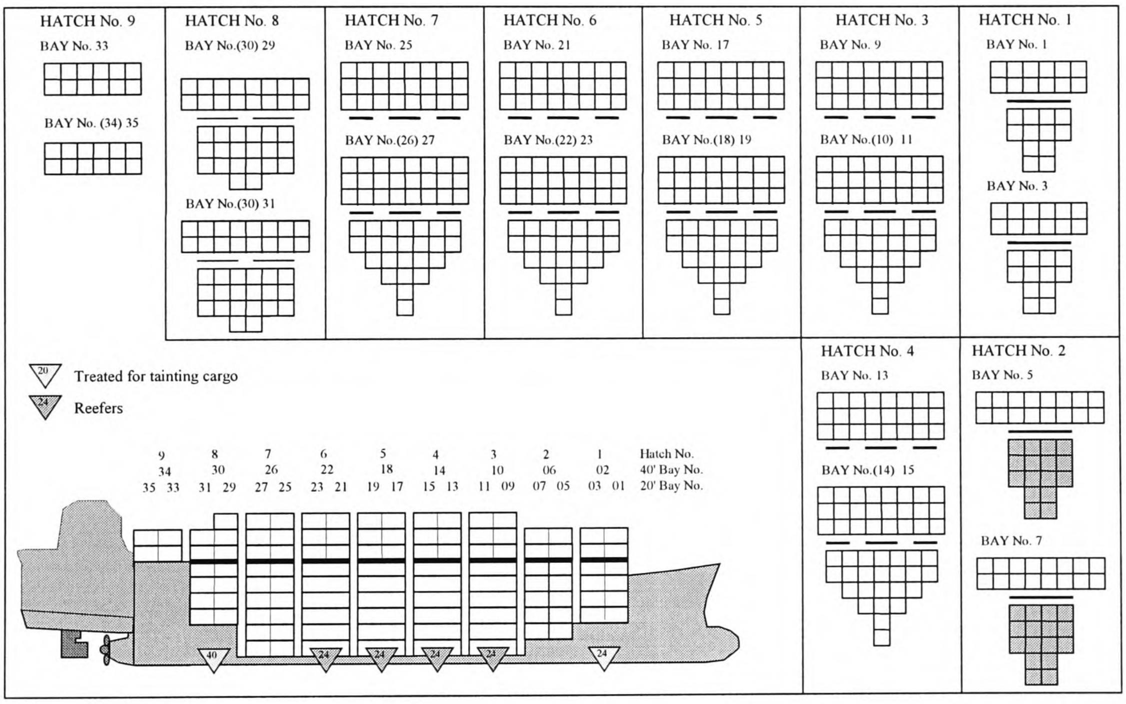


Figure 9-1 Outline Plan

For illustrating the methodology the following simplifications have been made:

- Two cranes are available for use at each of the ports considered (this being typical of many ports);

- Information regarding other features, such as the position of crew quarters, tanks and other vessel attributes is specific to a particular vessel and is, therefore, not included in the following evaluation;
- Constraints relating to intact stability are well documented in existing literature. Therefore, the following implementation and evaluation of the planning methodology excludes intact stability.

The described ship has an on-deck capacity of 352 TEU and an under-deck capacity of 336 TEU is for a total TEU capacity of 688. The loading and unloading strategies for four ports are considered with a total of 696 containers being loaded and 312 containers being unloaded for 1008 expected movements. Although the TEU capacity of the described ship is relatively small, the number of hatches under consideration during planning is representative.

Existing statistical information was used to generate typical percentages for the different lengths and types of containers used in this example, namely:

- 54% are 20' in length;
- 44% are 40' in length;
- 2% are of other lengths;
- 20% are reefers;
- 14% require special segregation;
- 66% are of a general type.

The container ship itself has been constructed with specific constraints upon where different lengths and types of containers can be stowed in order to represent constraints of a broad range of typical vessels:

- all on-deck bays can have 40' and 20' containers stowed there;
- under-deck hatches 1 and 8 can have 40' and 20' containers stowed there;
- under-deck hatches 2 and 7 can only have 20' containers stowed there;
- under-deck hatches 3, 4, 5 & 6 can only have 40' containers stowed there;
- under-deck hatches 1 and 8 are specially treated so that tainting cargo can be stowed there;
- under-deck hatches 3, 4, 5 & 6 can have reefers stowed there.

The next section describes how the stowage problem is decomposed, for this example, into smaller sub-problems that permit the effective application of search techniques.

9.4.4 Application of the planning methodology

The following sections describe how characteristics of the planning problem are modelled to produce the structures required for the application of search techniques in pursuit of stowage solutions. Each phase in the decomposition process is explained and related to the human planner's own approach to producing stowage solutions.

9.4.4.1 Longitudinal search

The objective of the longitudinal blocking phase in the pre-planning process is to:

- minimise the number of cargo spaces occupied by each destination;
- maximise the number of cranes in operation at each subsequent port.

A planner uses the General Arrangement document to assist in meeting the general stowage objectives. As the document's name suggests, the planner's task at this stage in the planning process is very general. Areas of the ship are 'reserved' to hold groups of containers with specific destinations. These destinations are usually marked on the General Arrangement using coloured pens. Given the highly generalised statistical information about future cargo available to the planner, little attention is given to placing specific containers at this stage in the planning process.

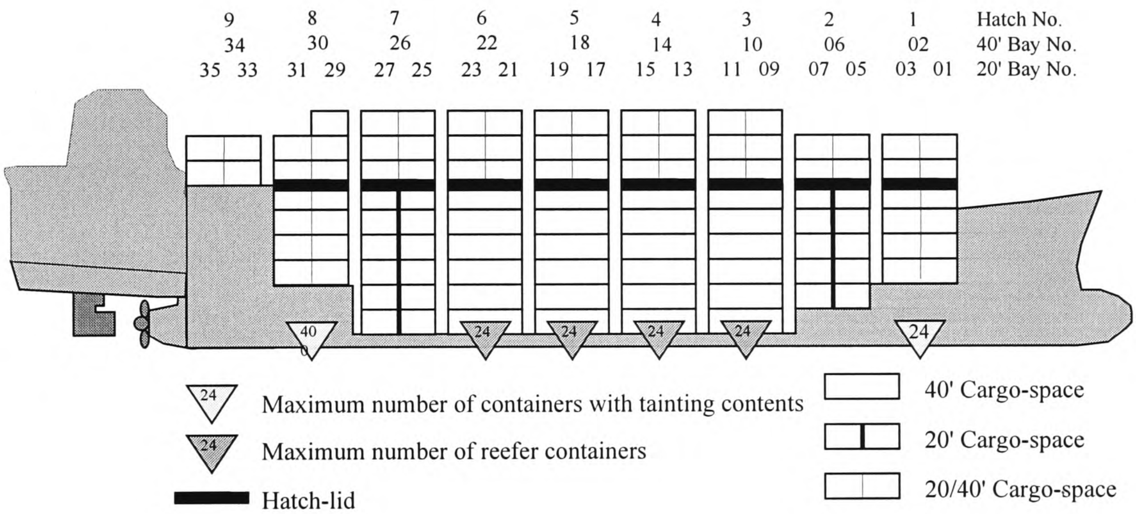


Figure 9-2 General Arrangement

Using the General Arrangement shown as part of the Outline Plan in Figure 9-1 (reproduced in Figure 9-2) as a model, a 'blocked' abstraction of the cargo-space was constructed (shown in Figure 9-3). The blocked 'General Arrangement' shows the human planner's conceptual view of the longitudinal cargo-space. Attention is focussed upon important ship characteristics, such as stowage capacities, stowage constraints and locations of hatch-lids and stowage considerations relating to crane efficiency. The blocked abstraction reduces the number of stowage locations under consideration from *n* slots to *x* blocks (in this case, from 568 slots to 17 blocks).

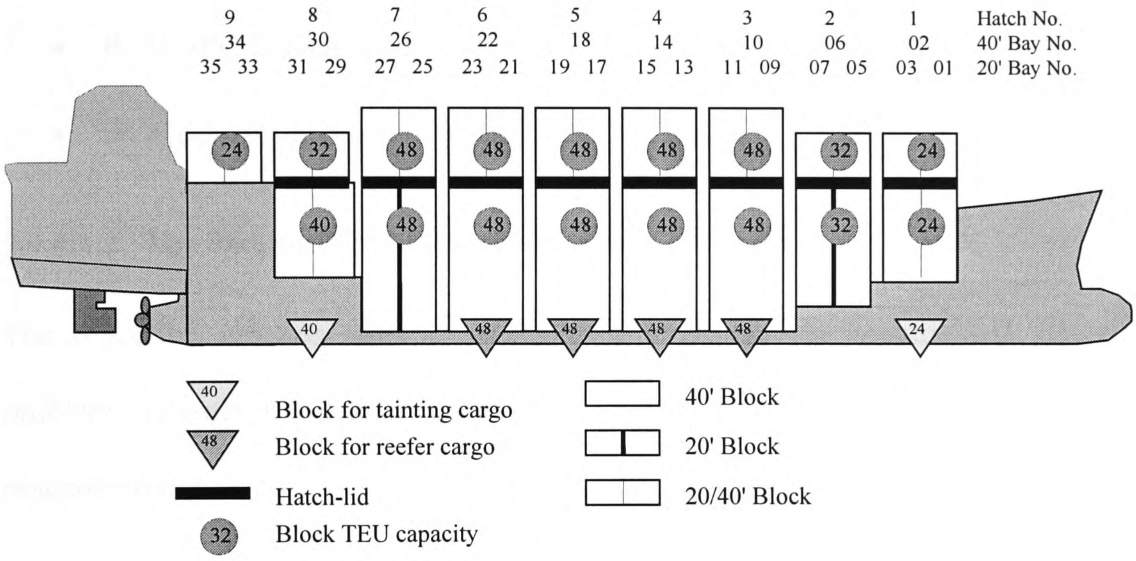


Figure 9-3 Blocked General Arrangement

9.4.4.1.1 The longitudinal abstraction

This section describes the underlying model for the longitudinal stowage problem. For the container loading and unloading process in the ports of the route under consideration, a number of problem characteristics required for producing viable solutions were identified. These were modelled as sets and functions applied to sets:

- $C:\{c_1...c_{nc}\}$ is the set of all containers;
- $P:\{p_1...p_{np}\}$ is the set of all ports of destination;
- $D:P \rightarrow \mathbb{P}C$ is each set of containers associated with each destination;

  e.g. $d_1:1 \rightarrow \{c_1,c_3,c_5\}$

  $d_2:2 \rightarrow \{c_2,c_4,c_6\}$

- $nd:\#\{\text{dom } D\}$ is the number of ports of destination;

- H: ℕ→ℙC is each set of containers associated with each hatch.

  e.g. $h_1 : 1 \rightarrow \{c_1, c_3\}$

- nh:#{dom H} is the number of hatches in the ship;
- B: ℕ→ℙC is each set of containers associated with each block.
- nb:#{dom B} is the number of blocks in the ship;

9.4.4.1.2 The longitudinal blocking objective function

The objective function used to evaluate solutions to the longitudinal stowage problem evaluates five aspects of a stowage pattern. The general expression of the objective function is:

$$f = (f_1 * w_1) + (f_2 * w_2) + (f_3 * w_3) + (f_4 * w_4) + (f_5 * w_5)$$

where f_i and w_i represent, respectively, an abstraction of the attractiveness of a solution and the weight, or importance, of that particular measure and goal is to minimise the returned value.

The first term of the objective function, f_1, which counts the number of blocks occupied by containers of each port of destination (POD). Minimising the number of blocks occupied by each POD assists in generating good block stowage.

$$f_1 = \sum_{i=1}^{nd} \sum_{j=1}^{nb} \begin{pmatrix} 1 \; if \left(\exists c : C | c \in \left(ran\left(d_i\right) \cap ran\left(b_j\right) \right) \right) \\ else \; 0 \end{pmatrix}$$

The second term of the objective function, f_2, counts how many hatches are occupied by containers of each POD and then compares this with how many cranes there are at that POD (in this case, 2). The objective here is ensuring that the number of cranes at a given POD is a factor of the number of hatches occupied by that POD.

$$f_2 = \sum_{i=1}^{nd}\left(\bmod\left(\sum_{j=1}^{nh}\begin{pmatrix}1\ if\ \exists c{:}C \mid \in \left(ran\left(d_i\right) \cap ran\left(h_j\right)\right)\\ else\ 0\end{pmatrix}\right)2\right)$$

The third term of the objective function, f_3, provides a measure of how well the containers are spread between hatches and, hence, how efficiently the cranes will be able to operate. Ideally, containers should be spread to allow all cranes to be used simultaneously throughout the unloading process.

$$f3 = \sum_{i=1}^{nd} ABS\left(\begin{array}{l}\left(Max\left(\forall_j{:}1..\#nh \mid count = \langle\rangle \bullet count' = count \wedge \#\{c{:}C \mid d_i \cap h_j\}\right)\right) - \\ \left(\sum_{k=1}^{length(count)-1}\left(\begin{array}{l}\left(\forall_j{:}1..\#nh \mid count = \langle\rangle \bullet count' = count \wedge \#\{c{:}C \mid d_i \cap h_j\}\right) \rhd \\ \left(Max\left(\forall_j{:}1..\#nh \mid count = \langle\rangle \bullet count' = count \wedge \#\{c{:}C \mid d_i \cap h_j\}\right)\right)\end{array}\right)\right)\end{array}\right)$$

The fourth term of the objective function, f_4, counts the number of POD that exist within each hatch. Minimising the number of hatches that are occupied by containers of each POD will lead to better block stowage.

$$f_4 = \sum_{i=1}^{nh}\sum_{j=1}^{nd}\begin{pmatrix}1\ if\left(\exists c{:}C \mid c \in \left(ran\left(d_i\right) \cap ran\left(h_j\right)\right)\right)\\ else\ 0\end{pmatrix}$$

The fifth term of the objective function, f_5, penalises stowage patterns in which containers of a particular destination are stowed inside two hatches and those hatches are adjacent (preventing the two cranes from working simultaneously).

$$f_5 = 1\ if\left(\begin{array}{l}\forall_i{:}1..\#nd \mid \forall_j{:}1..\#nh \mid \exists c{:}C \mid \left(\sum_{k=1}^{nh} 1\ if\ \exists c{:}C \mid c \in \left(ran\left(d_i\right) \cap ran\left(h_k\right)\right)\right) = 2 \wedge \\ c \in \left(\left(ran\left(d_i\right) \cap ran\left(h_j\right)\right) \cap \left(ran\left(d_{i+1}\right) \cap ran\left(h_{j+1}\right)\right)\right)\end{array}\right)$$

Altering the weight, w_i, associated with each function, f_i, will produce different, but acceptable, solutions.

9.4.4.1.3 Search applied to the longitudinal abstraction

The branch and bound approach is a very useful method for solving discrete optimisation, combinatorial optimisation and integer problems in general and, as will be shown, it is well suited to the blocked stowage problem. [77] For the longitudinal stowage problem, the Branch and Bound algorithm and related sub-procedures are specialised as follows.

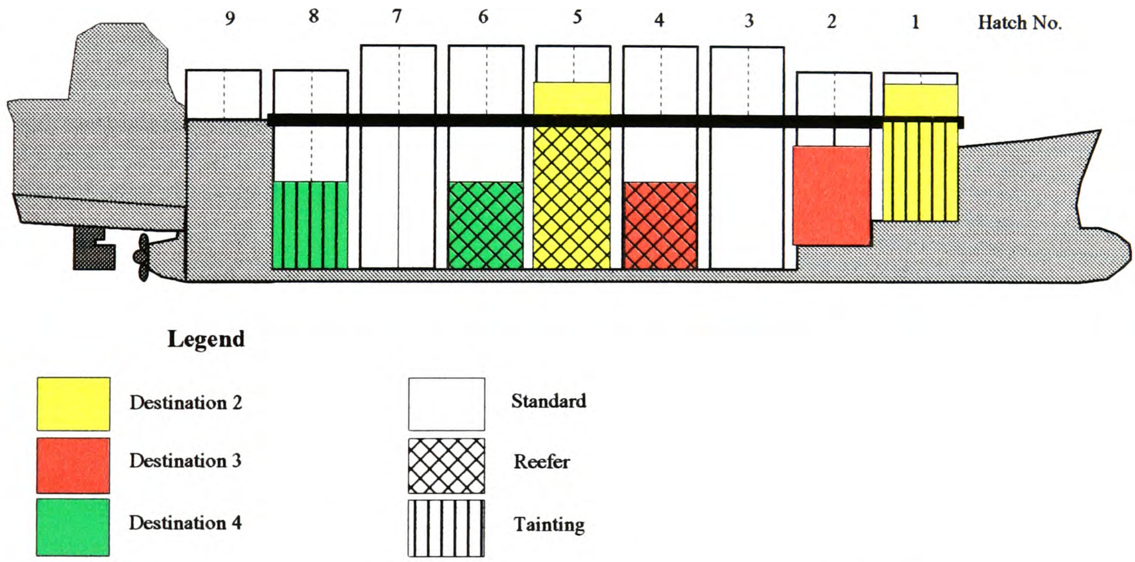


Figure 9-4 Discharged blocked stowage configuration

i. Initialisation

The initial state is made up of the discharged cargo-space, an ordered list comprised of all containers to be loaded at current port of call and an evaluation of the fitness of the stowage configuration. The discharged cargo-space illustrated in Figure 9-4 has 17 separate cargo-spaces to fill split between nine hatches. The list of containers to be loaded is ordered so that those containers for which there are fewest available locations (due to being tainting *etc.*) are placed first. Then, those groups of containers with the furthest POD are placed first in the sequence. The fitness of the solution reflects an abstract measure of the cost, based upon simulation of the unloading process at discharge ports, of the solution under consideration.

ii. Branching

New solutions are generated that reflect every possible placement into a block of the first container in the load-list associated with this candidate (seventeen, in the case of this example). All invalid solutions are then removed from the state-space. If after expanding a partial solution a feasible solution for the longitudinal stowage problem is found, then it is set aside.

iii. The search strategy

Each of the candidates produced during the branching process is sorted according to its fitness value and the number containers within its associated load-list. (Considering the length of the load-list rewards solutions where more have already been loaded). This strategy reflects a depth-first emphasis approach to the search process.

iv. Pruning

When one candidate sub-problem has the same fitness value as another but has more containers still to load then it can be deleted from the pool of partial solutions.

v. Choice of new sub-problem

The partial-solution with the best fitness value is selected as the new current candidate problem and the algorithm continues in a similar manner until n solutions are found. The number of solutions (n) generated at each POD is decreased in relation to how distant the POD is from the current port of call. (In the worked example, 4 solutions are generated at port 1 with 2 being passed on for consideration thereafter.) Upon termination of the search process (at which time we have n solutions for the current port) the problem is reinitialised and the process repeated for each of the n solutions. This process simulates a planning procedure at a given number of destination ports (4, in the case of this representative example). Once this process has been repeated for each destination, the best solution is the one with the least summation of the fitness values accumulated at each port.

9.4.4.1.4 Results

The described Branch and Bound algorithm was applied to the longitudinal blocking problem. The weightings for (w_i) that were applied to each function (f_i) for the described problem are given in Table 9.1. The longitudinal solution shown in Figure 9-5 was determined to be the best long-term solution upon considering the future implications at the next three ports.

Fitness (f_i)	f_1	f_2	f_3	f_4	f_5
Weighting (w_i)	2	4	3	1	10

Table 9.1 Weightings for longitudinal fitness functions

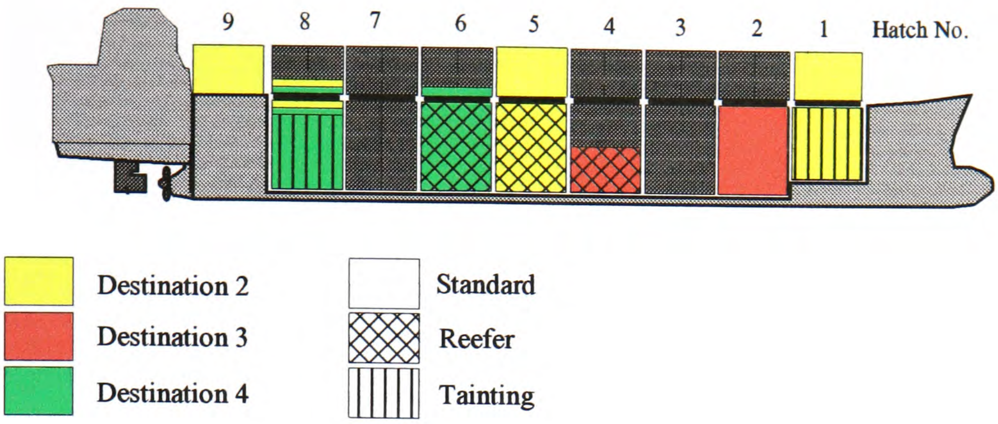


Figure 9-5 Outbound solution for port 1

The blocked General Arrangements for each of the subsequent solutions along the best path are shown in the following diagrams. Each diagram shows that the objectives identified in 9.4.4.1, that crane efficiency should be maximised and the number of stowage locations used be minimised, has been translated into practical solutions by the longitudinal blocking procedure.

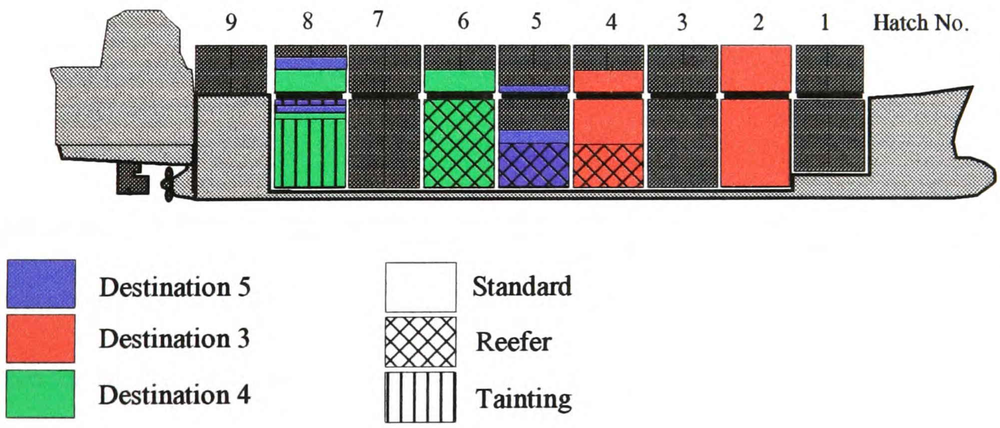


Figure 9-6 Outbound solution for port 2

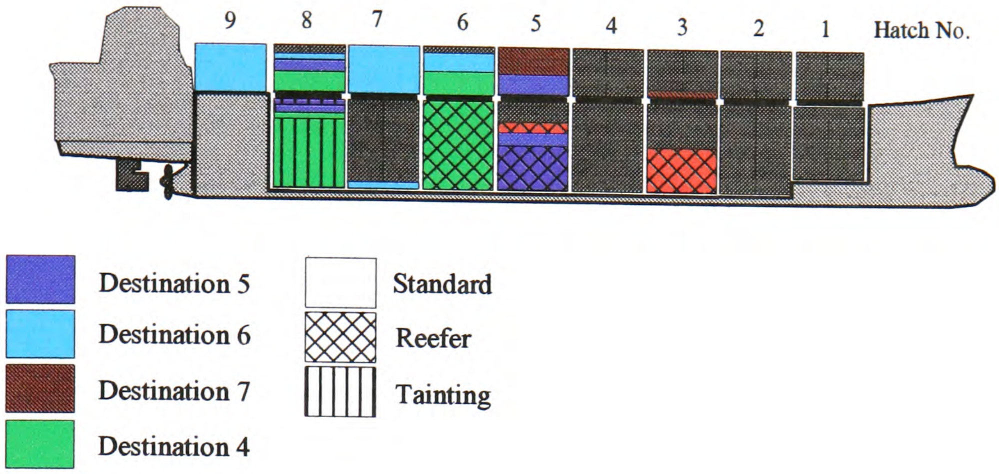


Figure 9-7 Outbound solution for port 3

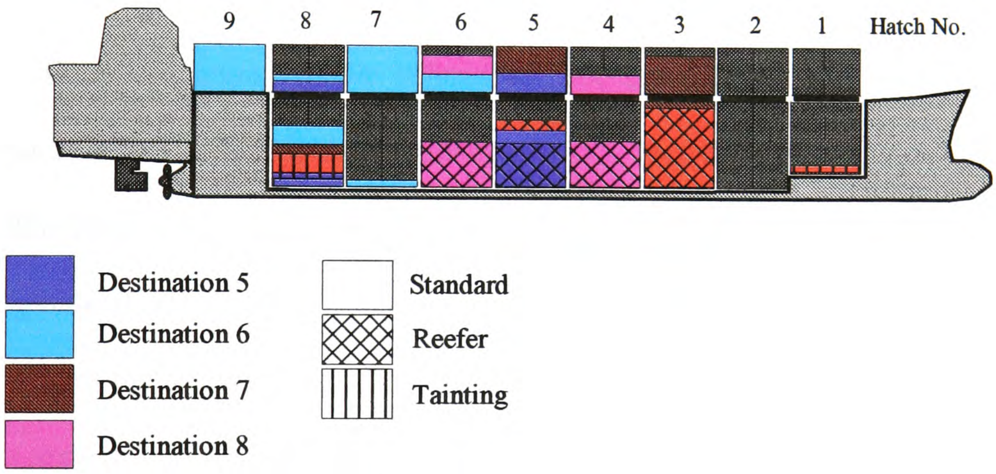


Figure 9-8 Outbound from port 4

For this example, four solutions were passed from port 1 to port 2, for further ports 2 solutions were passed on for consideration. Figure 9-9 shows how the combinatorial complexity of the stowage problem increases the further into the voyage solutions are considered (where the actual factor used for this experiment is compared with two alternative factors). Therefore, the author proposes that the number of solutions passed on to the next port for consideration should be reduced, by experimentation, to a number applicable to the time available and the trade route under consideration.

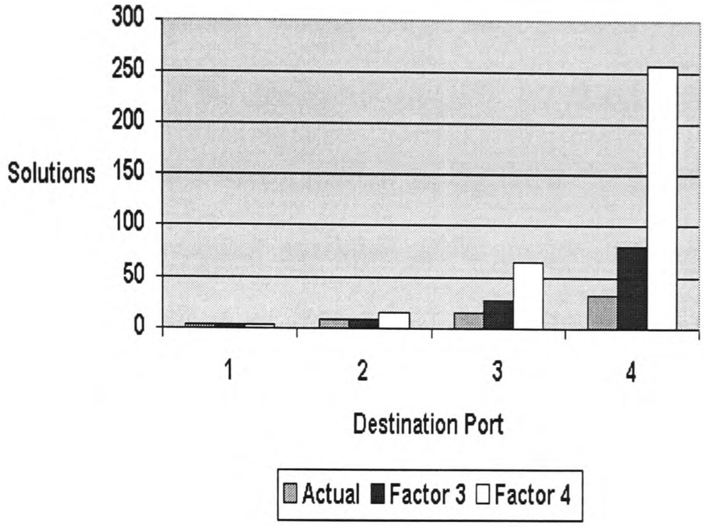


Figure 9-9 Combinatorial complexity of the multi-port problem

The fitness function can be refined to embody specific stowage requirements applicable to a particular trade route and vessel. Previous stowage decisions, visible within each of the depicted General Arrangements, influence the efficiency of the longitudinal placement procedure. Good block stowage at earlier ports greatly facilitates the search process for good solutions at later ports. The effectiveness of this approach to planning is dependent upon the quality of cargo forecasts and the ship operator's cargo acceptance practice. (*i.e.* the efficiency of the whole planning procedure is linked to how late a ship operator will accept cargo for transportation and how different from forecast information this cargo is). Approximately 4 minutes processing time was required to generate each of 60 solutions considered.

9.4.4.2 Latitudinal search

The objective at this point in the pre-planning process is to:

- minimise the number of hatch-lids moved;
- minimise the number of over-stows;
- minimise the number of cargo blocks occupied by containers.

To aid the planner in meeting these stowage objectives, a document called an Outline Plan is used. As the name of the document suggests, the planner's task at this stage in the planning process is to prepare outline instructions for the stowage of cargo. Given the generalised information available about cargo even a few days before docking, little attention is given to placing specific containers at this stage in the planning process. Instead, groups of containers of a general type and destination are allocated to groups of slots. This latitudinal planning procedure is modelled here by first making generalised placements, which are then specialised during the next phase of the planning procedure. During the longitudinal blocking procedure generally described containers were placed in blocks that correspond to specific hatches. In latitudinal blocking, the generally described containers are placed into sub-blocks within their assigned blocks. These sub-blocks, viewed latitudinally on an Outline Plan, are illustrated in Figure 9-10. This procedure is explained in the following sections.

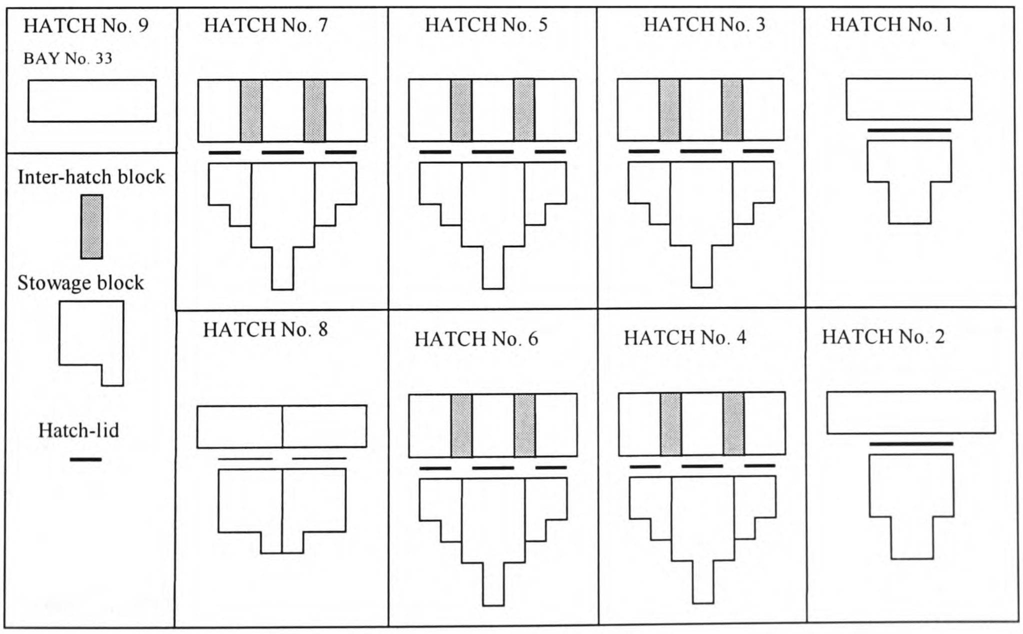


Figure 9-10 Blocked Outline Plan

9.4.4.2.1 The latitudinal abstraction

For the container loading and unloading process in the ports of the route under consideration, a number of problem characteristics required for producing viable solutions were identified, namely:

C: $\{c_1 \ldots c_{nc}\}$ is the set of all containers;

P: $\{p_1 \ldots p_{np}\}$ is the set of all ports of destination;

D: $P \rightarrow \mathbb{P}C$ is the set of containers mapped to each POD;

e.g. $d_1 : 1 \rightarrow \{c_1, c_3, c_5\}$

$d_2 : 2 \rightarrow \{c_2, c_4, c_6\}$

nd:#{dom D} is the number of POD;

H: $\mathbb{N}_1 \rightarrow \mathbb{P}C$ is the set of containers mapped to each stowage block;

e.g. $h_1 : 1 \rightarrow \{c_1, c_3\}$

nh: #{dom H} is the number of stowage locations;

B: $\mathbb{N}_1 \rightarrow \mathbb{P}C$ is the set of containers mapped to each block;

e.g. $b_1 : 1 \rightarrow \{c_5, c_6, c_7\}$

R: $\mathbb{N}_1 \rightarrow \mathbb{P}H$ is the set of blocks mapped to each block;

e.g. $r_1 : 1 \rightarrow \{6\}$

nr: #{dom R} is the number of blocks;

L: $\mathbb{N}_1 \rightarrow \mathbb{P}C$ is the set of containers stowed under each hatch-lid;

e.g. $l_1 : 1 \rightarrow \{c_5, c_6, c_7\}$

nl: #{dom L) is the number of lids;

max: $\mathbb{N}_1 \rightarrow \mathbb{N}_1$ is a function that returns the capacity of a block;

vol: $\mathbb{N}_l \rightarrow \mathbb{N}$ is a function that returns the volume of used space within a block;

9.4.4.2.2 The latitudinal blocking objective function

The objective function used to evaluate solutions to the latitudinal stowage problem considers four aspects of a stowage pattern. The general expression of the objective function is:

$$f = (f_6 * w_6) + (f_7 * w_7) + (f_8 * w_8) + (f_9 * w_9)$$

where f_i and w_i represent, respectively, an abstracted measure of the attractiveness of a solution and the weight, or importance, of that particular measure. Better solutions will return lower objective function values.

The first term of the objective function, f_6, counts the number of blocks occupied by containers of each POD. Minimising the number of blocks used is advantageous.

$$f_6 = \sum_{i=1}^{nd} \sum_{j=1}^{nb} \begin{pmatrix} 1 \; if \left(\exists c : C \mid c \in \left(ran\left(d_i\right) \cap ran\left(b_j\right)\right)\right) \\ else \; 0 \end{pmatrix}$$

The second term of the objective function, f_7, counts how many containers are stowed on hatch-lids, beneath which are containers destined for an earlier port. This particular type of overstow is thereby penalised by the function evaluation.

$$f_7 = \sum_{i=1}^{nb} \sum_{j=1}^{nb} \sum_{k=1}^{nc} \sum_{l=1}^{nc} \begin{pmatrix} 1 \; if \left(c_k \in \left(ran \, b_i\right)\right) \wedge \left(c_l \in \left(ran \, b_j\right)\right) \wedge \\ \left(\left(dom \, r_j\right) \in \left(ran \, r_i\right)\right) \bullet D^{\sim}[c_k] > D^{\sim}[c_l] \\ else \; 0 \end{pmatrix}$$

The third term of the objective function, f_8, provides a measure of how well the containers are spread between hatch-lids and, hence, how efficiently the cranes will be able to operate.

$$f_8 = \sum_{i=1}^{nc} \sum_{j=1}^{nd} \sum_{k=1}^{nl} \begin{pmatrix} 1 \; if \; c_i \in \left(ran\left(d_j\right) \cap \left(ran\left(l_k\right)\right)\right) \\ else \; 0 \end{pmatrix}$$

The fourth term of the objective function, f_9, counts how many empty spaces exist below a hatch-lid which supports containers. Such occurrences are indications of poor stowage, as these spaces are unavailable without first removing the hatch-lid and any containers stowed there-on.

$$f_9 = \sum_{i=1}^{nr} \sum_{j=1}^{nr} \begin{pmatrix} \left(vol\left(r_i\right) - \max\left(r_i\right)\right) \\ if \; \left(r_j \in ran\left(r_i\right)\right) \wedge \left(vol\left(r_j\right) > 0\right) \wedge \left(vol\left(r_i\right) < \max\left(r_i\right)\right) \end{pmatrix}$$

At this stage in the proposed system, all containers have been placed into a cargo-space (block). Therefore, the neighbourhood of possible moves under consideration has been reduced to the cargo within a solitary hatch.

9.4.4.2.3 Search applied to the latitudinal stowage problem

This section describes how branch and bound search is applied to the latitudinal blocked stowage problem. For the latitudinal stowage problem, the Branch and Bound algorithm and related sub-procedures are specialised as follows.

i. Initialisation

The initial state is comprised of a set of discharged cargo-spaces which correspond to hatch-lids within the hatch; an ordered list comprised of all containers allocated to that hatch during the longitudinal stowage procedure; and an evaluation of the its fitness. The cargo-space illustrated in Figure 9-5 has nine hatches of which seven require latitudinal stowage (corresponding to the hatches where cargo has been allocated during the longitudinal blocking procedure). The list of containers to be loaded is ordered so those containers with the fewest available stowage locations are first. Containers with the furthest POD are placed first within each of the sub-groups. The fitness of the solution reflects an abstract measure of the cost, which is based upon an evaluation of the unloading process at the discharge port under consideration. In addition to the hazardous and specialised constraints upon container placement, containers may only be placed within their on-deck or below-deck longitudinal blocks.

ii. Branching

New solutions are generated that reflect every possible placement of the first container within the latitudinal blocks associated with its assigned on-deck or below-deck longitudinal block. All invalid solutions are then removed from the state-space.

iii. The search strategy

Each of the candidates produced during the branching process is sorted according to its fitness value and the number containers within its associated load-list. This strategy reflects a best first approach to the search process.

iv. Pruning

Partial solutions with more containers to load and with a worse fitness value are pruned from the state-space.

v. Choice of new sub-problem

The best partial solution is selected as the new current candidate problem and the algorithm continues in a similar manner until the best found.

9.4.4.2.4 Results

Given the theoretical model under consideration, Figure 9-11 illustrates the resulting stowage configuration using the weightings for each function given in Table 9.2.

Function (f_i)	f_6	f_7	f_8	f_9
Weighting (w_i)	1	4	2	3

Table 9.2 Weightings given to the latitudinal fitness functions

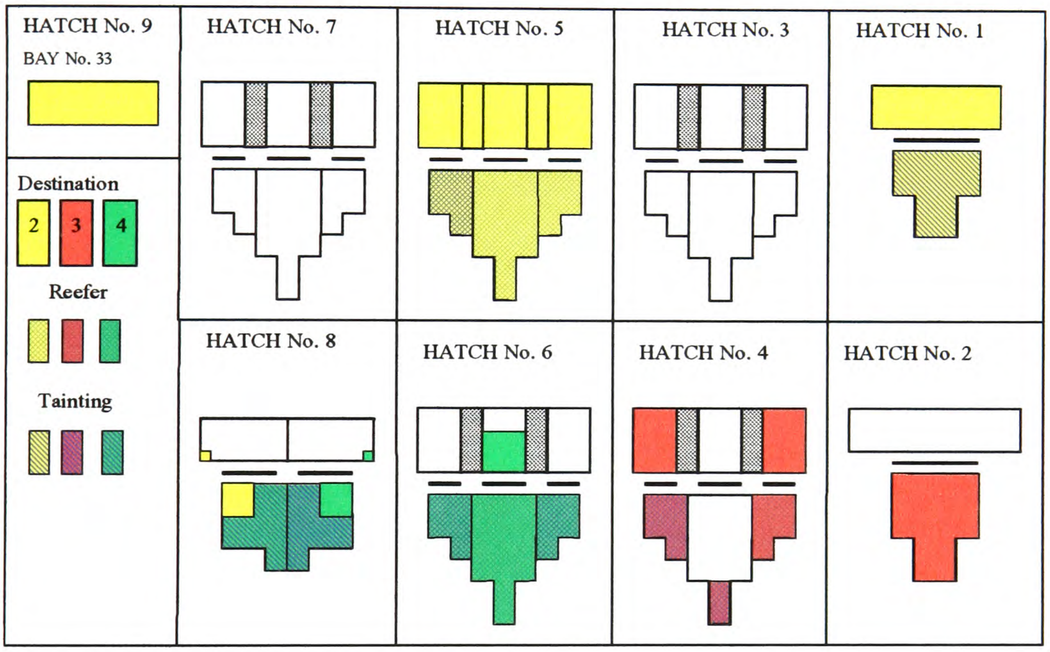


Figure 9-11 Latitudinally placed cargo

An examination of this diagram reveals that all containers have been distributed according to the objectives defined in Section 9.4.4.2. For example:

- Reefer and tainting cargo have been stowed appropriately;
- Crane splits have been maintained;
- No overstows exist;
- Hatch-lid manipulation has been minimised;
- Unused space below hatch-lids has been minimised.

Here, the planning is restricted to a single hatch at a single port. Approximately 2 minutes is required to plan the configuration for a single hatch. Having allocated all cargo according to the generalised stowage procedure outlined above, the next stage in the proposed planning methodology is to make specific allocations of containers to slots.

9.4.4.3 Bay-plan optimisation

A planner uses a document called the Bay Plan to make specific allocations of containers to stowage locations. Specific stowage locations are given to containers previously associated with a generalised area of the cargo-space of the ship. An adhesive label that precisely describes each container is attached to its corresponding slot on the Bay Plan. Information at this stage in the planning process is more specific, although the degree of specificity usually increases as the departure time gets closer. Now that all cargo has been allocated to a block within the cargo-space, the next step is to take each of these blocks and allocate specific stowage locations for each of the containers placed there. This is accomplished by adopting a two-stage procedure for planning the stowage configuration for each of the cargo blocks. Stage one of the proposed stowage procedure involves using heuristics to generate an initial stowage configuration. Stage-two of the proposed methodology involves using Tabu Search to optimise this initial configuration. The following implementation and evaluation of the bay-plan optimisation process deals only with the underlying heuristics used to evaluate the attractiveness of stowage solutions; ballast, for example, has been excluded. The following sections describe this proposed two-stage specialisation procedure.

9.4.4.3.1 Heuristically allocating containers to slots

The objective here is, simply, to prepare an initial starting point from which an optimum solution can then be determined during the next phase in the planning process. A randomly generated stowage configuration would permit the application of Tabu search to the problem in the next phase. However, using a sensible heuristic to generate a starting solution facilitates the stowage-configuration optimisation process. A number of heuristics can be used to pack the cargo-space, which reflect common-sense approaches to making specific container placements. The following describes a specific example of heuristically allocating containers to slots within a block using 3D packing theory applied to the representative worked example.

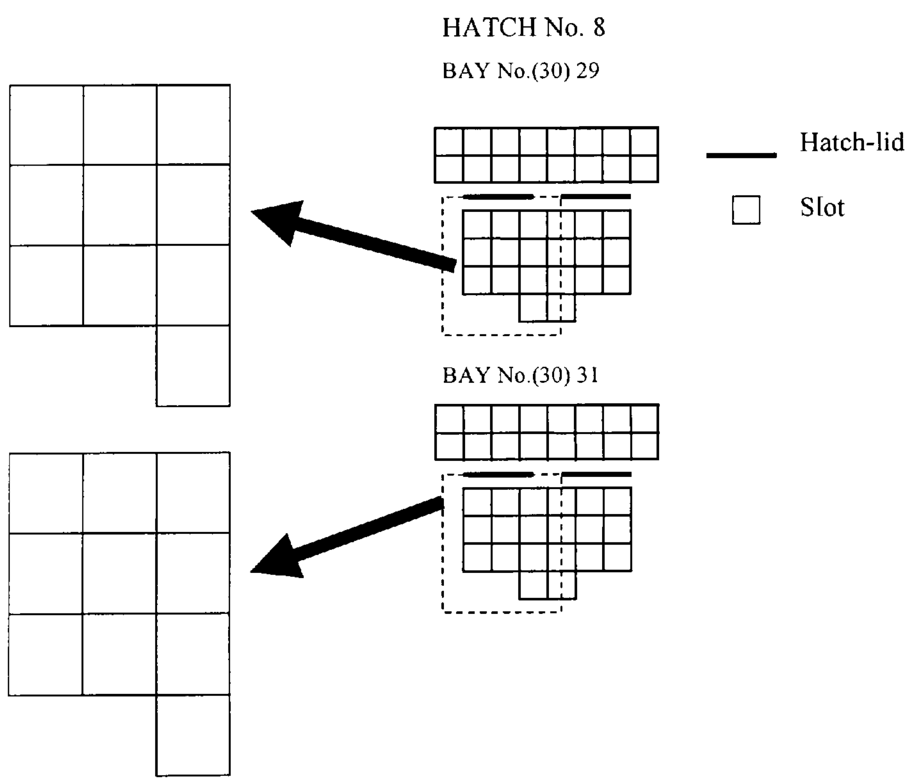


Figure 9-12 Bay-plan showing block to 'pack'

An example of a block that has containers to be allocated is shown in Figure 9-12 with the starboard under-deck block highlighted (20' bays 29 and 31 were combined to create a stowage block for the latitudinal blocking procedure). During the longitudinal and latitudinal blocking procedure, twenty containers were allocated to this block.

A variant of the 3D packing heuristics described in Chapter 8, designed to sequence containers into blocks, was used to generate an initial distribution of these twenty containers, namely:

1. The list of containers is ordered according to destination and weight.
2. The first container is taken from the list.
3. The container to be loaded is placed in the first available stowage location, searching each bay, stack and tier in the sequence shown in Figure 9-13.
4. If the list of containers is empty then the placement procedure is terminated, otherwise the process begins again at point 2.

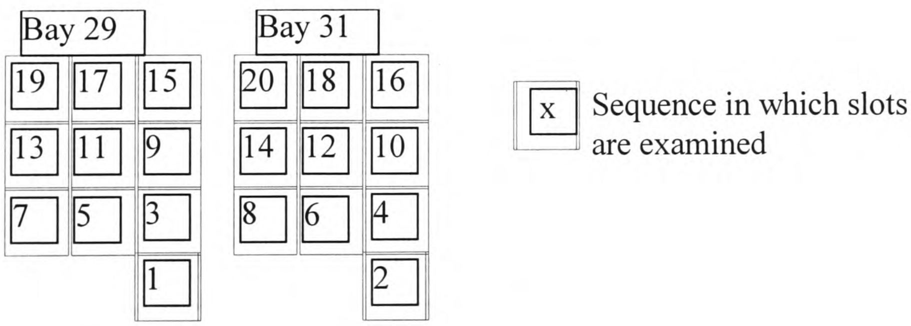


Figure 9-13 Order slots are filled

Applying this packing algorithm to the cargo-space and associated load-list resulted in the stowage configuration illustrated in Figure 9-14. This particular solution is near optimal and gives a good starting place for the next phase in the planning methodology.

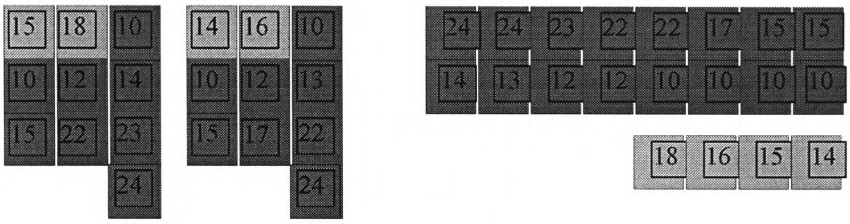


Figure 9-14 Example of a heuristically 'packed' block

9.4.4.3.2 Optimising the heuristically planned cargo-space

This section deals with the final phase in the stowage planning procedure, that of optimising the distribution of containers within each block of the Bay Plan.

9.4.4.3.2.1 Definitions for the fitness function

For the container loading and unloading process in the ports of the route under consideration, a number of problem characteristics required for producing viable solutions to the cargo-space optimisation problem were identified:

$I:\{c_1...c_{nc}\}$ is the set of all containers;

D_i is the destination port of container i;

DR_i is the set of restows associated with container i;

DW_i is the set of containers in the same stack stowed above container i with a weight greater than container i;

DS_i is the set of containers stacked with container i with a different POD.

9.4.4.3.2.2 Objective function for the block optimisation algorithm

Once a cargo space has been heuristically filled, or packed, the final step is to rearrange the containers such that:

- Restows are minimised;
- Container weight is graded upwards in the cargo space, heaviest to lightest;
- Stacks with mixed POD are minimised.

The general expression for the objective function for the container assignment within a block problem is:

$$f = (w_{10} * f_{10}) + (w_{11} * f_{11}) + (w_{12} * f_{12})$$

where w is the weighting associated with function f. A low value of f indicates a good stowage.

The first term of the objective function, f_{10}, counts the number of restows.

$$f_{10} = \sum_{i=1}^{nc} \sum_{j=1}^{nc} \begin{pmatrix} 1 \ if \ j \in DR_i \\ else \ 0 \end{pmatrix}$$

The second term of the objective function, f_{11}, counts the number of containers with a different POD stowed in the same stack.

$$f_{11} = \sum_{i=1}^{nc} \sum_{j=1}^{nc} \begin{pmatrix} 1 \ if \ j \in DS_i \\ else \ 0 \end{pmatrix}$$

The third term of the objective function, f_{12}, counts the number of containers with a greater weight stowed above each other in the same stack.

$$f_{12} = \sum_{i=1}^{nc} \sum_{j=1}^{nc} \begin{pmatrix} 1 \ if \ j \in DW_i \\ else \ 0 \end{pmatrix}$$

9.4.4.3.2.3 Container placement optimisation using Tabu search

Tabu search can be viewed as an iterative technique which explores a set of problem solutions by repeatedly making moves from one solution s to another solution s' located in the neighbourhood N(s) of s. These moves are performed with the aim of reaching a near optimal solution by the evaluation of some objective function $f(s)$ to be minimised.

To prevent the search process from returning a local optimum f, a guidance procedure is incorporated that accepts a move from s to s' even when $f(s')>f(s)$. Should no improving move be found in a given number of iterations then the local solution is returned as the global solution. This in itself could lead to cycling causing the process to return repeatedly to the same local solution without moving towards a global solution.

Tabu search circumvents the problem of cycling by preventing recent moves from reoccurring for a given number of iterations. Specifically, for each solution s a set of, legal, non-tabu moves m which can be applied to s in order to obtain a new solution $s' = s \oplus m$, giving $N(s) = \{s' \mid \exists\ m \in M(s)\}$. For the container to slot allocation problem, the neighbourhood $N(s)$ was determined by the longitudinal and latitudinal planning procedures described, respectively, in Section 9.4.4.1 and Section 9.4.4.2. That is, the neighbourhood would include only swaps of containers in the same latitudinally assigned blocks. The initial solution s has been determined by the heuristic placement of container within their neighbourhoods.

Given this, the form of the procedure used was as follows:

$s^*:=f(s)$, $k:=1$, $j:=1$

While $(j<\text{Max}\ (j)) \wedge (k<\text{Max}\ (k)) \wedge (f(s^*)\neq 0)$

$j:=j+1$

$M^* \subseteq N(s,k)$ (all legal, non-tabu, states)

Choose the best s' in M^*

$s:=s'$

If $f(s')<f(s^*)$ then $s^*:=s'$, $k:=1$ else $k:=k+1$.

End of While

Continuing with the example shown in Section 9.4.4.3.1, the heuristically filled cargo-space shown in Figure 9-15 was optimised using the above algorithm where $f(x)$ is described in Section 9.4.4.3.2.2. The optimised cargo-space is shown in Figure 9-16.

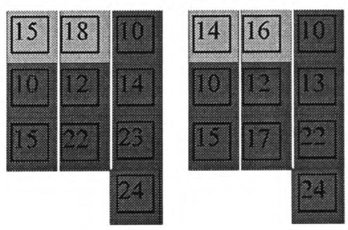


Figure 9-15 Heuristically filled cargo-space

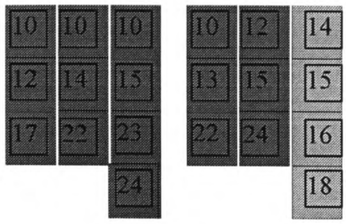


Figure 9-16 Optimised cargo-space

The weightings used for each of the fitness functions are given in Table 9.3. The optimised cargo-space shown in Figure 9-16 contains no restows, gradation of container weights within a stack is present and the mixture of POD with a stack has been minimised.

Function (f_i)	f_{10}	f_{11}	f_{12}
Weighting (w_i)	3	1	2

Table 9.3 Weightings associated with bay-plan optimisation functions

9.4.5 Conclusion

Results were obtained on a 166MHz Pentium with 40 megabytes of memory using Allegro Lisp to encode the blocking algorithms and GFA (a PC based 3GL with a high degree of functionality and graphical display features) to encode the container placement algorithm. The blocking process returned results within hours and the optimisation process, typically, requires a few seconds for each optimised block. Replacing the prototyping languages with optimally written code would result in a significant reduction in processing time. In all test problems theoretical data was used and this lead to the conclusion that a strong relationship between the shipping operator, container ship, route and market must exist in order efficient use of cargo-space be realised.

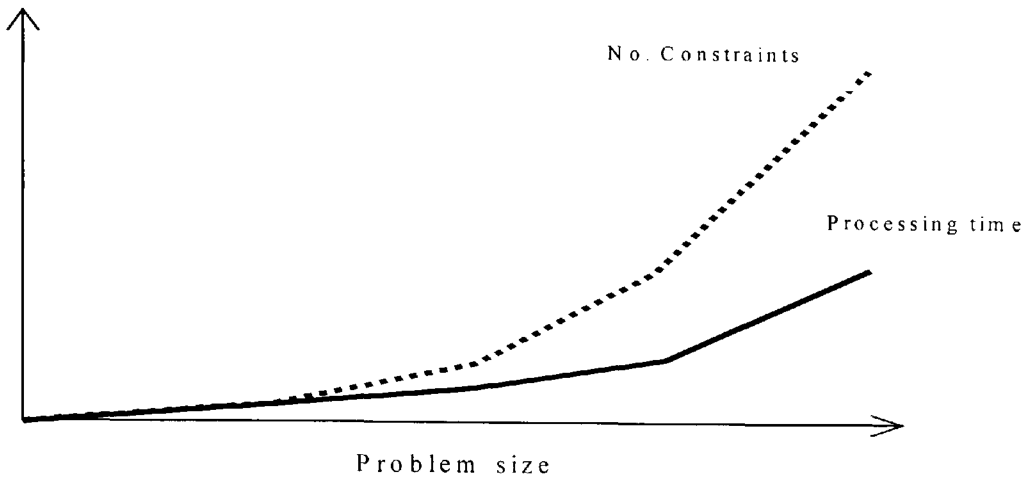


Figure 9-17 Expected increase in time in relation to the number of constraints

The number of valid branches, for a specific sub-problem, would be decreased in proportion to the number of constraints applicable and the type of cargo carried. A proportionate increase in the processing time required to process the constraints upon each solution would follow, but this is expected to ascend relatively gently (illustrated in Figure 9-9). Therefore, the algorithm is expected to be applicable in practice. The neighbourhoods under consideration when optimising cargo-spaces could be increased to include all blocks where containers belonging to the same POD are stowed. This could often result in a better global stowage configuration but would be dependant upon the planning method used by the ship operator planner (*i.e.* whether a complete stowage plan is prepared before loading starts).

9.5 Future Work

Although the basic stowage planning problem has been identified in this report, there is still much experimentation required in determining the exact data structures and search algorithms used (the latter being dependent on the former). More advanced search techniques such as Simulated Annealing [69] and Tabu Search [58,59,60,61,62] could then be experimented with. Experimentation with different data-structure representations of the domain is seen as the key to producing efficient solutions to the problem.

9.5.1 Simulated Annealing

Clustering of containers into like sets (such as those with like destination) where diminishing increments of these groups are loaded during the strategic planning phase (described in Section 8.2.3) offers an opportunity to explore Simulated Annealing. Clustering containers into groups that can be placed as a block will reduce the overall size of the state-space considerably and thereby speed up the search process.

9.5.2 Different neighbourhoods and Tabu Search

Varieties of different neighbourhood reductions are possible within the proposed optimisation phase of tactical planning. Experimentation within this project was limited to considering all containers within a cargo-space block. Experimentation with a number of different alternative neighbourhoods, such as considering only a particular class of container may offer overall improvements.

9.5.3 Geometric modelling of the cargo-space

Further modelling of the container-ship stowage space is complicated by the number of non-standard container configurations that preclude a simple cellular (array-like) representation of the three-dimensional space. An approach that places containers into a geometric space (in this case the bays of the container carrier) using packing algorithms has been looked at. However, a fuller experimental review of this approach would not only produce a suitable method of providing solutions for this domain, but could also be adapted for use in a variety of other similar problems.

9.5.4 Evaluation of stowage solutions

The demanding nature of the stowage-planning problem makes the determination of means of assessment difficult. Much time and consideration was given to setting-up procedures and criteria for evaluation for stowage solutions. To this end, meetings with individuals with relevant expertise were arranged. [63, 64] However, a more comprehensive study of evaluation criteria is needed which is based upon first hand experience of stowage planning at Tonnage Centres, where the actual planning is performed. The quality of any solution could then be appraised, generating solutions that are closer to the theoretical optimum.

9.5.5 The travelling salesman and the container-terminal

Additional factors, such as container and crane movement during the loading process, factored, or substituted, into the evaluation of a stowage-plan would result in container load sequencing that minimises cost from the container-terminals viewpoint. The container-terminal planners task has characteristics similar to the travelling salesman problem [73], where the 'salesman' in this case is the yard container transportation vehicle, and offers a significant opportunity to research a real-world problem in this context.

Bibliography

(1) **Perakis, A. N. & Dillingham, J. T.,** The Application of Artificial Intelligence Techniques in Marine Operations., The Society of Naval Architects and Marine Engineers, Paper presented at 1987 Ship Operations, Management and Economics International Symposium U.S.M.M.A., Kings Point, New York, Sept. 17-18, 1987.

(2) **Cho, D. W.**, Development of a Methodology for Containership Load Planning., Ph.D. dissertation, Oregon State University, Corvallis, Oregon, 1984.

(3) **Roach, D. K. & Thomas, B. J.,** Portworker development programme Unit C.2.2, Container-ship Stowage Plans, International Labour Organisation, ISBN 92-2-109271-2, 1994.

(4) **Martin, G. L., Randhawa, S. U. & McDowell, E. D.,** Computerised Container-ship Load Planning: A Methodology and Evaluation. Computers ind. Engng., Vol. 14, No. 4. pp. 429-440, 1988.

(5) Stowage and Segregation Guide to IMDG- Code. Published by U.O. Storck Verlag (Stahltwrek 7, D-2000 Hamburg 50, Germany).

(6) **Boden, J.,** Private communication. Maritime Computer Technical Services Ltd., Cardiff Business Technology Centre, Senghennydd Road, Cardiff, CF2 4AY.

(7) **Goldberg, L. L.,** Principles of Naval Architecture. J. P. Comstock, Chapter II, Intact Stability, pp. 63-142, 1980.

(8) **van Hee, K. M. & Wijbrands, R. J.**, Decision support system for container terminal planning. European Journal of Operational Research no. 34, pp. 262-272, 1988.

(9) **Kockumation, A. B.**, Capstan: linking computer-aided stowage to EDI. World Freight Technology '92, Sterling Publications International Ltd., ISSN 0965 1977.

(10) **Tiemroth, E. C.**, SPARCS: a revolution in terminal operations. World Freight Technology '93, Sterling Publications International Ltd., ISSN 0965 1977.

(11) **Soudan, L.**, SEAGHA - A private sector network for the electronic exchange of goods and transport data in the Port of Antwerp. HINTERLAND Quarterly review, XXXVI, 1987, ISSN 0773 -1922.

(12) Private communication with G. Ross, Maritime Computer and Technical Services.

(13) **Shields, J. J.**, Container-ship Stowage: A Computer-Aided Preplanning System. Marine Technology, Vol. 21, No. 4, pp. 370-383, 1984.

(14) Review of Maritime Transport ,1993. United Nations Conference on Trade and Development. United Nations publication. ISBN 92-1-112292-9. ISSN 0566-7682.

(15) **Inglis, I.**, Unpublished set of lecture notes, University of Glamorgan, Department of Computer Studies, 1993.

(16) **Winston, Patrick Henry**, Artificial Intelligence, Third Edition, publisher-Addison Wesley, 1992.

(17) **Barr, Avron, Feigenbaum, E. A., Cohen, P. R.**, The Handbook of Artificial Intelligence (Three Volumes), William Kaufman, Los Altos, CA, 1981.

(18) **Rich, E.**, Artificial Intelligence, McGraw-Hill, New York, 1983.

(19) **Patterson, D. W.**, Introduction to Artificial Intelligence and Expert Systems. Englewood Cliffs, N.J., Prentice Hall, 1990.

(20) **Boden, M.A.**, Artificial Intelligence and Natural Man. London, MIT Press, 1987.

(21) **Alvarado, S. J., Dyer, M. G., & Flowers, M.**, Knowledge Engineering, Volume I, Fundamentals. McGraw-Hill Publishing Company, Chapter 10, pp. 286-344, 1990.

(22) **Schank, R. C. & Riesbeck, C. K.**, Inside Computer Understanding, five programs plus miniatures, Hillsdale, New Jersey, Lawrence Erlbaum Associates, 1981.

(23) **Schank, R. C. & Abelson, R. P.**, Scripts, Plans, Goals and Understanding. Hillsdale, NJ: Lawrence Erlbaum Associates, 1977.

(24) **Kodratoff, Y.**, Introduction to Machine Learning, Pitman, London, 1988.

(25) **Kodratoff, Y.**, Knowledge Engineering, Volume I, Fundamentals. McGraw-Hill Publishing Company, Chapter 8, pp. 226-255, 1990.

(26) **Michalski, R. S., Carbonell, J. G. & Mitchell T. M.**, Machine Learning: An Artificial Intelligence Approach, Morgan Kaufmann, Los Altos, Calif, 1983.

(27) **Quinlan, J. R.**, Learning efficient classification procedures and their application to chess end games. Machine Learning: An Artificial Intelligence Approach, Morgan Kaufmann, Los Altos, Calif, pp. 463-482, 1983.

(28) **Harandi, M. T. & Lange, R.**, Knowledge Engineering, Volume I, Fundamentals. McGraw-Hill Publishing Company, Chapter 4, pp. 103-129, 1990.

(29) **Schank, R. C., & Leake, D. B.**, Creativity and Learning in a Case-Based Explainer, Artificial Intelligence, vol. 40, 1989.

(30) **Lenat, D. B. & Brown, J. S.**, Why AM and EURISKO Appear to Work, Artificial Intelligence, vol. 23, 1984.

(31) **Furse, E.**, Unpublished set of lecture notes, University of Glamorgan, Department of Computer Studies, 1993.

(32) **Beynon-Davies, P.**, Expert Database Systems. A Gentle Introduction, published by The McGraw-Hill International Series in Software Engineering, 1991.

(33) **Golshani, F.**, Knowledge Engineering, Volume I, Fundamentals. McGraw-Hill Publishing Company, Chapter 2, pp. 28-51, 1990.

(34) **Jackson, P.**, Introduction to Expert Systems, Addison-Wesley, 1986.

(35) **Waterman, D. A.**, A Guide to Expert Systems. Addison-Wesley Publishing Company, Reading, Mass., 1986.

(36) **Botter, R. C. & Brinati, M. A.**, Stowage container planning: a model for getting an optimal solution. IFIP Transactions B (Applications in Technology), vol. B-5, pp. 217-29, 1992.

(37) **Dowsland, K. A. & Dowsland, W. B.**, Packing Problems. European Journal of Operational Research, vol. 56, no.1, pp. 2-14, 1992.

(38) **Ivanic, N., Mathur, K. & Mohanty, B. B.**, An Integer Programming Based Heuristic Approach to the Three-dimensional Packing Problem. J. Mfg. Oper. Mgt. 2, pp. 268-298, 1989.

(39) **Mazouz, A. K.**, Heuristic Techniques Application in a 3-D Space. SPIE Vol. 1008, Expert Robots for Industrial Use, 1988.

(40) **Maruyama, K., Chang, S. K., Tang, D. T.**, A General Packing Algorithm for Multidimensional Resource Requirements. International Journal of Computer and Information Sciences, Vol. 6, No. 2, 1977.

(41) **Han, C. P., Knott, K., Egbelu, P. J.**, A Heuristic Approach to the Three Dimensional Cargo Loading Problem. Computers & Industrial Engineering, vol.11, no. 1-4, pp. 109-13, 1986.

(42) **George, J. A. & Robinson, D. F.**, A heuristic for packing boxes into a container. Comput. & Ops Res. Vol. 7, pp. 147-156, Pergamon Press Ltd., 1980.

(43) **Saginaw, D. J. & Perakis, A. N.**, A Decision Support System for Container-ship Stowage Planning. Marine Technology, Vol. 26, No. 1, pp. 47-61, 1989.

(44) **Lang, G. J. P.**, Some Computer Aids in the Loading of Deep Sea Container Vessels, - A Personal Experience. Proceedings, Computer Applications in Operation and Management of Ships and Cargoes, conference sponsored by RINA, London, 1985.

(45) **Sansen, H.**, Ship-Planner, A Conception Support Tool for the Bay Plans of Container Ships., Systemia, Domaine de ST Hilaire, Pichaury 13290 AIX LES MILLES, FRANCE, 1989.

(46) **Sato, K., Itoh, H. and Awashima, Y.**, Expert System for Oil Tanker Loading/Unloading Operation Planning. Ship and Offshore Engineering Department, Ishikawajima-Harima Heavy Industries Co., Ltd., 2-1-1, Toyosu, Koto-ku, Tokyo 135, Japan. Computer Applications in the Automation of Shipyard Operation and Ship Design, VII - C. Barauna Vieira et al. (Eds.) Elsevier Science Publishers B.V. (North Holland), 1992 IFIP.

(47) **Mizukami, A.**, Loading Planning System for Oil Tankers Applied with Artificial Intelligence Techniques, Bulletin of Soc. of Naval Architects of Japan, No. 740, 1991.

(48) **Mortimer, A. J.**, Information Structure design for databases, a practical guide to data modelling. Oxford, Butterworth-Heinemann, 1993.

(49) **Dillingham, J. T. & Perakis, A. N.**, Application of Artificial Intelligence in the Marine Industry. Fleet Management Technology Conference, Boston, Mass., April 1986.

(50) **Crainic, T. G., Gendreau, M., Soriano, P. & Toulouse**, A tabu search procedure for multicommodity location/allocation with balancing requirements. Annals of Operations Research, volume 41, pp. 359-383, 1993.

(51) **Reeves, C.R. & Beasley, J.E.**, Modern Heuristic Techniques for Combinatorial Problems. Published by: McGraw Hill Book Company Europe, 1995

(52) **Dillingham, J.**, Design of an expert system for container storage planning. Fleet management technology conference, 1987.

(53) **Hartman, P.J.**, Practical Applications of Artificial Intelligence in Naval Engineering. Naval Engineers Journal, November 1988.

(54) **Dowsland, K.A. & Dowsland, W.B.**, Packing problems. European Journal of Operational Research, volume 56, number 1, pages 2-14, 1992

(55) **Han, C.P., Knott, K. & Egbelu, P.J.**, A heuristic approach to the three dimensional cargo loading problem. International Journal of Production Research, volume 27, number 5, pages 757-74, May 1989.

(56) **Maruyama, K., Chang, S. K., Tang, D. T.**, A general packing algorithm for multi-dimensional resource requirements. International Journal of Computer and Information Sciences, Vol. 6, No. 2, 1977.

(57) **George, J. A. & Robinson, D. F.**, A heuristic for packing boxes into a container. Comp. & Ops Res. Vol. 7, pp.147-156, Pergamon Press Ltd., 1980.

(58) **Glover, F.**, Tabu Search: A Tutorial. Interfaces. Vol. 20, pp.74-94, Aug. 1990.

(59) **Glover, F.**, Tabu Search-Part I. Operations Research Society of America Journal on Computing. Volume 1, No. 3, pp. 190-206, 1989.

(60) **Glover, F.**, Tabu Search-Part II. Operations Research Society of America Journal on Computing. Volume 2, No. 1, pp. 5-32, 1990.

(61) **Glover, F.**, A user's guide to tabu search. Annals of Operations Research, Volume 41, pp. 3-28, 1993.

(62) **Glover, F. & Laguna, M.**, Tabu Search. Modern Heuristic Techniques for Combinatorial Problems, McGraw-Hill Book Company, Chapter 3, pages 70-150, 1995.

(63) Technical information provided by The Maritime Computer & Technical Services Ltd., Cathays, Cardiff.

(64) Technical information provided by P&O Containers Limited, Beagle House, Braham Street, London, E1 8EP.

(65) **King, Todd.** Dynamic data structures : theory and application. - San Diego; London : Academic Press, 1992.

(66) **Tremblay, Jean-Paul.** An introduction to data structures with applications, 2nd ed. - New York; London : McGraw Hill, 1984.

(67) **Wirth, Niklaus.** Algorithms and data structures. London: Prentice-Hall, 1986.

(68) **Willis, C.J.,** Private communication. P&O Containers Limited, Beagle House, Braham Street, London, E1 8EP.

(69) **Dowsland, K.A.,** Simulated Annealing, Modern Heuristic Techniques for Combinatorial Problems, Chapter 2, Pages 20-63, McGraw-Hill Book Company, 1995.

(70) Report by the United Nations Conference on Trade and Development secretariat, published on the World Wide Web at address: http:www.unicc.org/unctad/en/pressref/mt3dus2.htm, November 1993.

(71) **King, T.,** Dynamic data structures : theory and application. Academic Press, 1992.

(72) **Winston, P. H.,** LISP. 3rd ed., Addison-Wesley, 1989.

(73) **Reeves, C. R.,** Modern Heuristic Techniques for Combinatorial Problems. McGraw-Hill Book Company, 1995.

(74) **Smith, M. F.,** Software prototyping : adoption, practice and management. McGraw-Hill, 1991.

Appendix A

A.1 Modelling cargo and cargo-space relationships

Container-ships can be seen as consisting of Bays, Stacks and Tiers. Combining these entities gives a location of a single cell. Cells can be either on deck or below deck. Cells below deck are accessed via hatch-lids. Cells on deck are removed to reveal the hatch-covers needed to access cells below deck. Bays are grouped below deck into compartments. Compartments are separated by bulkheads. Using the above description an entity relationship modelling exercise provides an initial step towards understanding how best to physically model a container-carrier and the cargo loaded upon it. An analyse of this description (where the perceived entities are underlined) is diagramatically reproduced below.

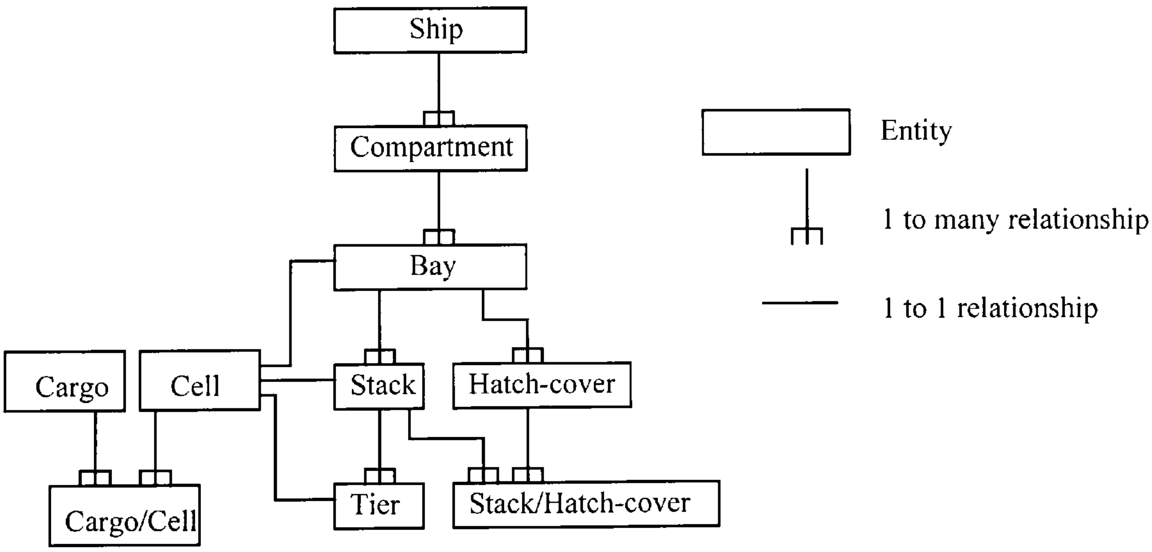


Figure A-1 Cargo-space and cargo relationship diagram

The diagram is interpreted and relationships noted so that a complete picture of the model can be determined. These relationships are shown in the following table.

Relationship	**Comments**
Ship/Compartment	A ship has many compartments
Compartment/Ship	A compartment belongs to one ship
Compartment/Bay	A compartment can have many bays
Bay/Compartment	A bay belongs to one compartment
Bay/Stack	A bay can have many stacks
Stack/Bay	A stack belongs to one bay
Bay/Hatch-cover	A bay can have many hatch-covers
Hatch/Bay	A hatch-cover belongs to one bay
Hatch/Stack	A hatch-cover can cover many stacks
Stack/Hatch-cover	A stack can have many hatch-covers
Stack/Tier	A stack can have many tiers
Tier/Stack	A tier belongs to one stack
Cell/Bay-Stack-Tier	A cell is composed of a bay, stack & tier
Cell/Cargo	A cell can contain many items of cargo
Cargo/Cell	Cargo can occupy many cells

Figure A-2 Relationships

Appendix B

B.1 LISP as a problem solving programming language

The principle data type used by LISP is the list, hence the name of the programming language. A list is simply a series of elements or atoms. Searching a state-space typically involves the processing and combining of each element contained within a list (search is described in Section 4.1). Given that the source state contains only one node, the possible number of new nodes is equal to the number of available moves, or, in the case of the deep-sea container-ship stowage problem, the number of moves from a stowage configuration is equal to the number of empty stowage locations. The following section describes how search is used to explore different combinations of, in this case, paths. The example given is a general search problem that illustrates well characteristics of search.

B.2 Shortest path problem

A typical example of how search can be used to solve a problem is where the shortest path must be found from point S, source, to point G, goal, by traversing a number of connected nodes, where connections between the nodes have a distance associated with them (illustrated in Figure B.1). Encoding and solving this type of search problem using LISP illustrates a typical approach taken.

Appendix B

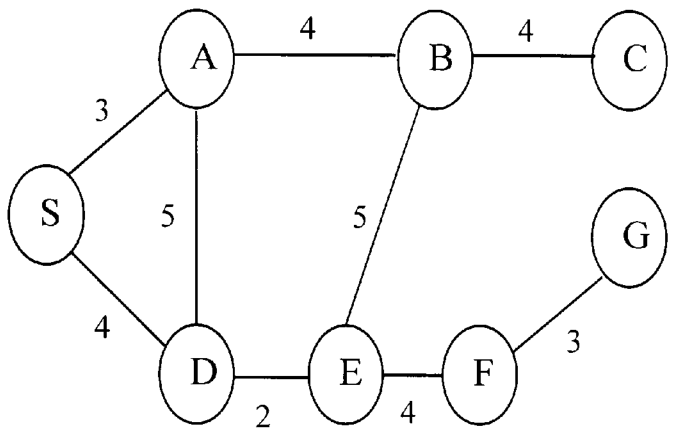


Figure B.1 Example of a directed graph (network)

Search procedures explore networks such as the one illustrated, learning about the connections between nodes and their associated distances as the network is processed. One way of representing how search finds a solution to this problem is with a search tree (explained in Section 4.1.1). How LISP uses this concept is best explained by working through a complete example. The following illustrates how lists can be used to represent a search tree and how LISP processes this representation.

B.3 Encoding the directed graph using LISP

Without explaining the exact code used, the following introduces key concepts associated with representing and solving state-space problems using LISP. Each of the arcs between the nodes shown in Figure B.1 represent the distance between two points. The distance between each of the nodes is stored symbolically using the following form:

(Distance (S A)) returns 3 and

(Distance (A B)) returns 4.

The neighbours of a node are similarly stored using the following form:

(Neighbour S) returns (A D) and

(Neighbour B) returns (A C E).

A data-structure stores node adjacency information, so that each of the neighbours for each node can be found. Another data-structure stores the distances associated with each link between nodes.

B.4 Encoding the state-space using LISP

A generic representation for the source node that enables the state of the problem to be developed and similarly stored is described here. Since the objective is to find the shortest path between node S and node G, a record of the path and distance travelled will provide a suitable basis for a representation. One representation for a source node is a list containing two elements, path and distance, *e.g.* ((S) 0). The important thing to note here is the use of brackets to separate elements of a list.

B.5 Exhaustively searching the state-space using LISP

Given the source state ((S) 0) the next step is to expand this node creating a number of sibling nodes. To do this, an algorithm is implemented, using LISP, which recursively removes the first element from the list, and expands it. Initially, only the source node is in the list, and so the source node will be removed and expanded. The neighbours for the node under consideration are determined along with the associated distance. Expanding the source node in this way creates two new nodes, made up of a path and a distance travelled, that are concatenated together to yield a new list, *i.e.* (((S A) 3) ((S D) 4)). This new list now represents the current state of the problem (illustrated in Figure B.2).

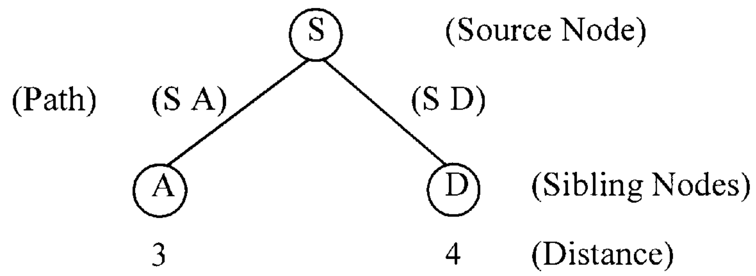


Figure B.2 Search tree representing the state of the problem after two moves

The current state of the problem now contains two elements, each representing a path and distance. Since the state-space has not yet been completely explored the current state is itself expanded. This process will continue recursively until all possible paths have been exhausted. Processing this list now yields a new list of elements expanded from both the nodes contained within the current state. This expansion yields a state that describes the paths taken (((S A B) 7) ((S A D) 8) ((S D A) 9) ((S D E) 6)).

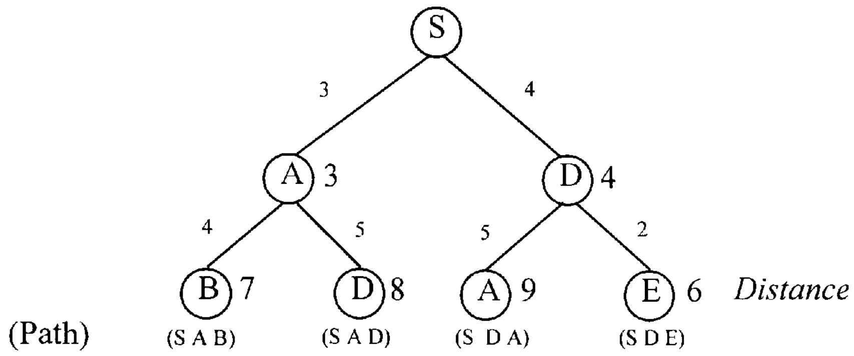


Figure B.3 State of the problem after six moves

One of the key features of LISP is that all data are stored identically, as a list. The difference between elements stored within a list, *e.g.* characters and numbers, is decided during processing. This gives a very flexible way of representing different data types and structures. The current state of the problem now contains four elements or paths (see Figure B.3).

Continuing this process of concatenating the expanded elements of the current state to create a new state (illustrated in Figure B.4) yields (((S A B C) 11) ((S A D E) 12) ((S A B E) 10) ((S D A B) 13) ((S D E B) 11) ((S D E F) 10)). Paths that return to the start, are called cyclic, and are removed since they are redundant.

An option often taken is to search the elements in the list for the best path found so far (in this case the path with the minimum distance travelled). Expanding the best node first would often permit the early cessation of the search, as discussed in

Sections 4.2.2 and 4.2.3, but for ease of understanding the problem is exhaustively processed without sorting.

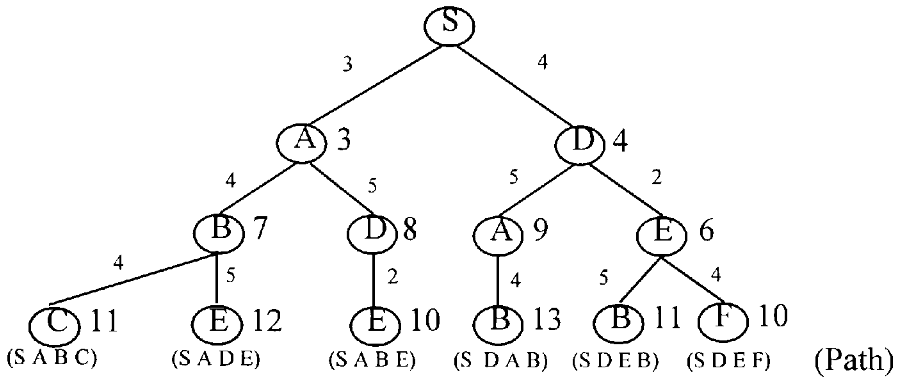


Figure B.4 State of the problem after twelve moves

Continuing this process finishes with a number of destination states representing all the possible expanded paths, with cyclical paths removed (illustrated in Figure B.5). The destination states would be represented by a list thus, (((S A B E F G) 19) ((S A D E F G) 16) ((S D A B E F G) 21) ((S D E F G) 13)). A quick inspection shows that path (S D E F G) is the shortest, with a total distance travelled of 13.

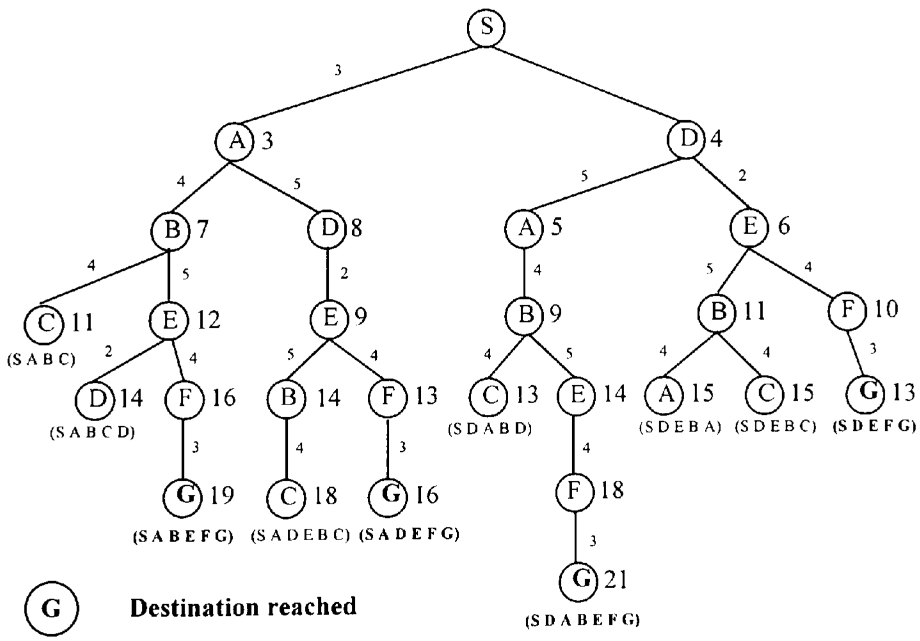


Figure B.5 Final state-space

Working through this example has served to illustrate how a state-space is encoded and traversed using LISP. This type of exercise is well documented in general texts[72] and has been used to solve a variety of optimisation problems.

Appendix C

C.1 Containerisation and standards

This Appendix outlines the present status and the modifications made to the International Standards Organisation (ISO) standard 668 and the situation in the field of use of non-ISO containers in different regions of the world. This Appendix also describes the impact of the introduction of larger containers on the different links of the multimodal transport chain based on surveys conducted by different international bodies. Finally, information about the dimensional characteristics of new containers entering into use is given. Much of the information that follows has been taken from the report on the United Nations Conference on Trade and Development (UNCTAD) by the secretariat (1993). [70]

C.2 Changing container dimension standards

Since its introduction, the ISO 668 standard for the containers has undergone constant update, despite an indication to the contrary by the ISO/TC 104 (the body entrusted with the work on container standardisation) in 1978 where it was recognised that "work on the standardisation of freight containers had reached the stage at which a high degree of stability can be maintained" (Resolution 71 of the 10th meeting of ISO/TC 104, see Annex 1). [70] Fortunately, none of these modifications has affected the intermodality (width and length) of containers, so that the 8' wide and 20' and 40' lengths, adopted at the beginning of standardisation, remain.

Appendix C

The most recent addition to accepted container dimensions is the 9' 6" high container (or high-cube). These high-cube containers are gaining popularity, making up more than seven percent of the world container population in 1992 [70] (see Table C.1). However, the inland transportation of such containers raise many problems (such as encountering low bridges) in many places. The high-cube continues to be treated as a special shipping operators and container lessors, clearly indicating that they are not treated as "standards". For this reason many countries were reluctant to accept the high-cube as an international standard.

	1980	1983	1986	1990	1992
Number of 9'6"x20' and 40' containers	41160	86366	188509	310050	457155
Total number of containers	2499456	3798730	4778866	5102563	6373051
Share of total, per cent	1.65	2.27	3.94	6.07	7.17
Growth rate	1.00	2.09	4.57	7.53	11.10

Table C.1 Increase of 20' and 40' long 9'6" high containers (in TEUs)

Note: For 1990 and 1992 only dry-freight containers (excluding thermal, tank, flatrack and other types of special containers) were taken into account.

The world container population has increased fourfold since 1978 and its homogeneity has been steadily growing thus evidencing the world-wide acceptance of the ISO standard (see Table C.2). Around 90 per cent of the world container population is now represented by 20' and 40' long containers with 8' 6" height.

	1980	1982	1986	1990	1992
Total number	2499456	3798730	4778866	5874004	7320400
8’6”x20’	934481	1934280	2185102	2475725	3157339
8’6”x40’	945668	1417338	2026636	2542952	3290166
8’6”=20’ and 40’	1880149	3351618	4211738	5018677	6447505
Per cent	75.2	85.6	88.1	85.4	88.0

Table C.2 Share of 20' and 40' long 8' 6" high containers (in TEUs)

C.3 Situation in the field of use of non-ISO containers

Apart from some domestic (United States) and regional trades (Europe), non-ISO containers are, in general, in use only in specific international trades. The share of such containers in the world container population is negligible. The markets in which such containers are in demand are becoming saturated and the orders for such containers are consequently declining.

According to a survey conducted by the International Association of Ports and Harbours (IAPH) in 1991 [70] out of a total 154 responding ports a third (49) reported having processed non-ISO containers. A total number of such containers - 1,541,039 - constituted only 3.8 per cent of the total number of container handlings reported (40,849,311). [70] These non-ISO containers included some high-cubes (56 per cent of total non-ISO reported). Of the 49 ports responding as having handled non-ISO containers, 26 ports (53 per cent) handled less than 1 per cent non-ISO containers, 15 ports (31 per cent) between 1 and 10 per cent, and 8 ports (16 per cent) between 10 and 100 per cent. [ibid] From its survey the International Association of Ports and Harbours (IAPH) has drawn the conclusion that because the handling of non-ISO containers remains confined to a certain range of ports, their handling cannot yet be identified as a global issue to be tackled as a whole by the world port community. [70]

Region	Number of responding ports	Number of non-ISO containers handled	Share, per cent
Africa	5	0	-
South America	1	0	-
Australasia	14	14	-
Caribbean	2	1709	0.1
Mid-East	14	2618	0.2
Europe	53	252299	16.4
Asia	46	439217	28.5
North America	19	845182	54.8
Total	154	1541039	100

Table C.3 Regional Distribution of non-ISO Containers Handled

In terms of geographical distribution (shown in Table C.3) most of the non-ISO standard containers are concentrated in the North America region, followed by Asia and Europe. The conclusion that most developing countries do not handle non-ISO containers is confirmed by the census of the world container population carried out by Cargoware International in mid-1992. [70] According to this census North American owners control 72.5 per cent of all non-ISO length containers. According to the 1992 census of the world container population conducted by Cargoware International [ibid], about 80 per cent of all high-cube inventories (9' 6" height) were held by owners in North America and the Far East.

Length	Carriers	Lessors	Total	Share, percent
53’	1200	-	1200	*
48’	7282	-	7282	*
45’	21263	997	22260	1.21
40’	214051	637744	851795	46.12
24’	12438	-	12438	*
23’	-	101	101	*
20’	69191	382636	951827	51.54
10’	-	48	48	*
Total	325425	1521526	1848951	100
TEUs	580717	2160507	2741224	

Table C.4 Composition of United States container population, 1991

Note: * Negligible percentage.

In spite of the rapid increase of domestic intermodal traffic, most revenue for operators of double-stack trains in North America has come from international maritime ISO containers, which represent the overwhelming majority of the container population in the United States (see Table C.4). The height of container is no longer limited to 9' 6" (2.9 m). To meet the requirements of certain categories of domestic shippers, American President Companies (APC) introduced a prototype of a new lightweight 53’ long, 9' 9" high, 8’ 6” wide container which is described as the largest intermodal container in the world.

The use of non-ISO containers is not a monopoly of United States domestic transportation. Swap-bodies and inland containers with different lengths and with a width of 2.5m are in general use in the European logistics systems in conjunction with palletised goods. It is estimated that an equivalent of about 200,000 TEUs of such units are presently in operation. Among them the so-called "cellular pallet-wide containers" (CPC) should be mentioned as gaining popularity in short-sea and coastal trades. [70]

For the purposes of the second Economic Commission for Europe (ECE) Seminar on the Impact of Increasing Dimensions of Loading Units on Combined Transport, the European Co-operation in the field of Scientific and Technical Research (COST) carried out a world-wide survey on the consequences of an introduction of proposed new generation containers, the so called COST 315 study. [70] The study revealed that at present new generation containers are unacceptable for Europe. [ibid]

C.4 New containers entering the industry

In 1992, the one million TEUs production level was passed for the first time with the figure standing at 1.15 million TEUs. [70] However, in the second half of 1992 the demand for new containers had slumped and, consequently, container production changed radically. The container manufacturers curtailed their production and sometimes closed their just-commissioned facilities. [ibid]

Table C.5 shows that the bulk of containers produced was represented by standard dry freight 20' and 40' containers, with the share of 40' containers in steady rise. Contrary to this, there is no growth in number of longer-than-ISO containers produced. This would seem to confirm that there is saturation of trades where such containers are in use.

Type of container	1992	1991	1990
Dry freight	1080000	870000	750000
Standard	1020000	810000	700000
Special*	44000	42000	25000
Euro pallet-wide	8000	8000	10000
(over 40' long)	8000	10000	15000
Refrigerated	64000	35000	44000
Tank containers	6000	5000	5000
World total	1150000	910000	800000

Table C.5 Estimated world container production by type (in TEUs)

Note: *Including open-top, flatrack, platform, ventilated, bulk, open-side types.

A survey conducted by the UNCTAD secretariat on the dimensions of dry freight containers entering the industry revealed statistics relating to length (shown in Table C.6) and height (shown in Table C.7). Among non-ISO lengths only 45' long containers were noticeable in the replies received by the UNCTAD secretariat (125 units). [70] The vast majority of the "others" in relation to the length was represented by 30' containers for dry bulk cargo, and the rest were 10' small containers.[ibid]

Length	1989 TEUs/Share	1990 TEUs/Share	1991 TEUs/Share	1992 TEUs/Share
20'	76209/60.6	68899/52.9	89904/43.0	132598/43.5
40'	49569/39.4	78998/47.0	119218/57.0	172366/56.5
45'	-	-	-	125***/**
Others*	-	214/0.1	-	-
Total	125778/100	168111/100	209122/100	305089/100

Table C.6 Distribution of containers entering the industry by length (in TEUs)

Note:

* Includes 10' and 30' containers. ** Negligible percentage. *** Real units.

In relation to height, containers other than 8', 8'06" and 9'06", except for 1992, were almost equally represented by half-height 4' 3" and 9' swap bodies. In the figures for 1992 all the "others" in relation to the height were represented by 9' height swap-bodies. [70]

Height	1989 TEUs	Share	1990 TEUs	Share	1991 TEUs	Share	1992 TEUs	Share
8'	6244	5.0	2155	1.3	219	0.1	-	-
8'6"	119304	94.8	165347	98.4	204497	97.8	291104	95.4
9'6"	-	-	-	-	3790	1.8	12985	4.3
Others*	230	0.2	609	0.3	616	0.3	1000	0.3
Total	125778	100	168111	100	289122	100	305689	100

Table C.7 Distribution of containers entering the industry by height (in TEUs)

Note: * Includes 9'0" and half-height (4'03") containers.

As was the case of the previous study in this respect [70] the survey did not reveal any significant signs of proliferation of non-ISO dimensions (length and height) among new containers. Contrary to this, the survey confirmed the trend towards the increasing of the demand in 40' containers and a significant consolidation of the position of 8' 6" high containers. High-cube 9' 6"-high containers are steadily taking their place in the production lines since 1991. With the adoption of this height as an international standard, the trend to their further spread will certainly be confirmed.

The Deep-sea Container-ship Stowage Problem: Modelling and Automating the Human Planning Process

I. D. Wilson, P. A. Roach

Division of Computing and Mathematics, University of Glamorgan
Pontypridd, Mid Glamorgan, CF37 1DL, United Kingdom

Abstract - Container-ships are vessels possessing an internal structure that facilitates the handling of containerised cargo. At each port along the vessel's journey, containers destined for that port are unloaded, and additional containers destined for subsequent ports are loaded. Determining a viable configuration of containers that facilitates this process, in a cost-effective way, constitutes the deep-sea container-ship stowage problem. The work of determining a stowage configuration for a container-ship, on leaving a port, is performed by human stowage planners, who work under strict time constraints, and are limited in the number of configurations that they can consider. Little work has been published in the area of full automation of stowage planning. Authors proposing full automation have correctly identified the salient features of the problem, but have failed to recognize how human planners solve the problem, instead allowing the array-like nature of spaces within containerised vessels to entirely dictate their approach to a solution. To enable implementation of these approaches, excessively large search spaces are pruned through the removal of important features of the problem, rendering the solutions not commercially viable. This paper proposes an approach which can determine good sub-optimal solutions to the entire problem in a commercially viable duration of time. This is achieved through an intelligent analysis of the domain allowing the problem to be divided into sub-problems, each of which may be solved through the application of search. Further, this approach allows many more stowage configurations to be considered than would be possible for a human planner.

1. Introduction & background

A container is a box that comes in a variety of dimensions and types that facilitates the transportation of cargo. The standardization of containers[6] has enabled the introduction of dedicated container carriers within an *inter-modal* transportation system. In particular, dedicated *cellular* container-ships (having an internal designed for handling standard-length containers) have become the norm in world-wide maritime services. Container transportation by sea involves the interaction of two commercial bodies, the *container-terminal operator* (who is responsible for loading and unloading of containers) and the *shipping operator* (who is responsible for transporting containers by sea). The work described in this paper reflects primarily the needs of the *shipping operator* in determining good stowage patterns which maximise vessel utilisation and minimise time in port.

The container - Each container is labelled with its own uniquely identifying code, part of which indicates the dimensions of a container. The International Standards Organisation (ISO) recommend that a container should be 8 feet wide, 8 feet 6 inches high and, most commonly, 20' and 40' long.[10] Containers not conforming to an ISO classification are termed *out-of-gauge* (over-width and/or over-height). Also, some containers are frames of standard dimensions which allow cargo to protrude. So-called *special* containers are designed for cargo requiring special handling (perhaps needing a power supply to either cool or heat contents). Certain types of special cargo are defined as *hazardous*.

Container-ship space geometry - Each *cell* of a cargo-space is considered to be 20' long, 8' wide and 4'3" high. The relationship between a cell and a physical *location* for a given container need not be one-to-one. Cells are grouped into vertical *stacks*, which are in turn grouped into *bays* (collections of stacks across the width of a ship). Bays are either on-deck, or below-deck (enclosed within the ship beneath removable hatch-lids), and are grouped together by associated hatch number (indicated in Figure 1). Below-deck bays have restrictions on lengths of containers which can be accommodated. The position of cargo within this cargo-space can be identified by a unique address. The term used to indicate how many containers of a standard height and width, twenty feet in length, a container-ship can carry is *Twenty-foot Equivalent Units* (TEU).

Container-loading and unloading - Containers are loaded from the bottom of the ship, into vertical stacks. All containers placed on a hatch cover (on-deck) must be removed, along with the hatch cover, to access containers below deck. Containers may need to be moved to allow access to ones to be unloaded, or simply to improve the overall stowage pattern - such movements are called *re-handles*. Containers blocking access to others to be unloaded first are known as *over-stows*. There are rules governing requirements to segregate hazardous cargo, at specified distances, from each other and certain cargo.

Ship stability - Part of the cargo stowage planner's task is to ensure, via *intact stability* calculations, that the vessel remains in a stable condition. Cargo weight should be spread evenly to avoid *heeling* (an inclination from the vertical towards port or starboard) and ensure close to zero *trim* (which reflects the angle of the vessel fore to aft).[5] Uneven weight distributions also produce forces which can distort the physical structure of the ship, namely *bending* (acting from bow to stern) and *torsion* (port to starboard).[5] Ballast (sea-water) may be used to stabilise a vessel, but counts as additional cargo and so should be kept to a minimum.

Stowage objectives

The planner must determine optimum placement of containers so that all constraints (restrictions upon where and how containers can be stowed) are satisfied and material handling costs (those associated with the loading and unloading of cargo) are minimised. The task is split into two main parts - the generation of long-term (generalised) and short-term (specialised) stowage strategies. The are termed by the authors, respectively, the *strategic* and *tactical* phases of stowage planning.

The difficulties involved in planning are magnified by the multi-port nature of the problem, requiring the planner to take into account the consequences that one port's plan will have at subsequent ports. Planners must also consider expected loads at subsequent ports, which often include statistical information describing loads in generic terms. The main constraints and guidelines[10] to be followed are:

- Minimisation of the number of container re-handles and over-stows, and of ballast;
- Maximisation of vessel utilisation, and crane utilization at the terminal;
- Correct segregation of hazardous cargo;
- Placement of out-of-gauge containers so as to minimise interference with adjacent locations;
- Appropriate placement of specials (*e.g.* next to power supplies);
- Intact stability requirements must be met;
- Maximum stack heights and stack weights should not be exceeded, and heavier containers should generally be placed below lighter ones;

2. Existing work

This real-world problem is one that would benefit from automation through the application of Artificial Intelligence. Many decision support systems exist that assist planners with the tactical phase of planning, automating the time-consuming calculations for ship stability. Also, considerable attention has been given to automating container-terminal processes. However, little work has been published in the area of full automation of stowage planning. Authors proposing full automation have correctly identified the salient features of the problem, but have allowed the array-like nature of spaces within containerised vessels to entirely dictate their approach to a solution. In these approaches, 'intelligence' is provided solely through the use of variants of search to addressing the placements of specific containers to specific cells. To enable the implementation of these approaches, excessively large search spaces are pruned through the removal of important features of the problem, rendering the solution not commercially viable. By concentrating solely on the specific placements of containers, these authors have not recognised how human planners solve the problem. The failure of this approach has, in the view of this paper's authors, resulted in a drying-up of research into the stowage problem.

Of particular interest are the Computer Aided Pre-planning System (CAPS) due to Shields[9] and work of Botter.[1] Shields treats the cargo-space as an array, and employs a weighted random approach to placing containers within that array. The weights are governed by sensible stowage criteria. CAPS is reported as reducing the number of over-stows. However, other shipping companies that have augmented or replaced paper-based stowage planning systems with computer-assisted methods also report similar improvements, without fully automating stowage planning.[7]

Botter provides a mathematical model for describing the entire stowage problem. This model also assumes an array-based approach to cargo-space. Botter reports that the model can, in theory, be used to find an optimal solution. However, the model demonstrates the computational complexity of the problem; Botter identifies the problem as being NP-Hard.[1] As an optimal solution can not be found for commercial ships in a reasonable processing time, Botter developed integer-programming methods, based on the mathematical model, which solve the problem only by ignoring important features.

Importantly, out-of-gauge and other special cargo are not included in the models due to Shields and Botter. This reduces the search space of the problem by ignoring important factors. However, both authors group containers with the same characteristics (such as destination port) prior to loading. Further, Shields uses general descriptions of the groups, so that generically described containers of a class, rather than precise containers, can be loaded. The approach proposed in this paper has built on this type of grouping and abstraction, to better model the processes used by human planners.

3. Approach proposed

An analysis of the documents used by human planners (the General Arrangement, Outline Plan and Bay Plan described below) revealed different conceptual levels of planning, from the general (or strategic) to the specific (or tactical).

3.1 Strategic planning documents

The *General Arrangement* document (illustrated in Figure 1) is a simplified, small-scale, vertical longitudinal section through the centre of the vessel, viewed from the starboard side.

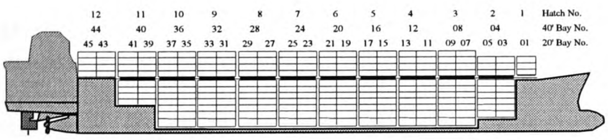


Figure 1 General Arrangement

This document provides information that can help when planning the ship operation, in particular: the location of each hatch (cranes may not be able operate simultaneously on bays located side-by-side); the position of the accommodation block and engine room (important when considering crane positioning and hazardous container stowage); bay restrictions on the lengths of containers which can be accommodated and on locations which can hold only empty containers. However, the General Arrangement does not show how many containers can be stowed across the vessel at each level above and below deck. That information is provided in an *Outline Plan* (a part of which is illustrated in Figure 2). Here, the container stowage stacks of the entire ship are shown in more detail, in the form of a series of vertical transverse sections, or bays, viewed from aft. Each stowage location is shown as a small box. The Outline Plan shows exactly how many containers can be stowed in each bay. Container positions are marked using letters and/or colours to indicate the container's port of discharge. Container slots can be marked with symbols to show the presence of overheight and over-width containers. It also allows indication of the positions of power supplies, and presence and type of any special and hazardous cargo. The planner can see at a glance how many hatch-covers (marked as thick black lines) will have to be removed before under-deck containers can be moved. The planner can also see how many above-deck containers will have to be removed before a hatch cover can be accessed.

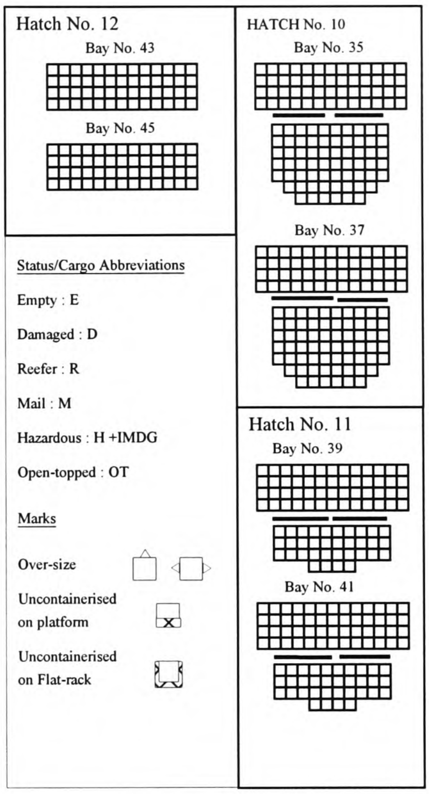


Figure 2 Outline Plan (part)

3.2 Strategic planning approach

In the strategic planning phase, a human planner considers placing containers into approximate positions in the cargo-space rather than necessarily in specific cell locations. The processes used by human planners in this phase are modelled through a new approach to computerised planning based upon the abstraction of the cargo-space. This abstraction reduces the solution space of the problem by allowing groups of containers to be assigned to spaces which are less specific than precise cell locations.

The three-dimensional cargo space is abstracted by blocking together groups of container locations. Stacks of cells sharing a common relationship to a hatch-lid are blocked (illustrated in Figure 3). This abstraction reduces the number of stowage locations considered when placing a container in the strategic planning phase (typically from approximately 2000 to approximately 100).

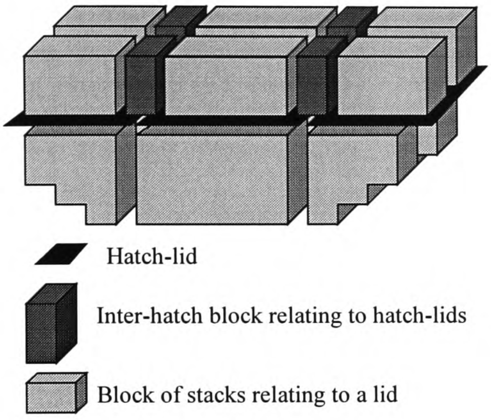


Figure 3 Example of cargo-space blocking (shown for part of a container-ship)

Each block is a three-dimensional space with a corresponding TEU container capacity. These can be filled by generically described groups of containers of varying dimensions without requiring assignment of specific containers to specific locations. Intact stress and stability can be calculated for the abstract model to an acceptable degree (using an approach based upon the work of Sato *et al* [8]). Consideration of planners' use of the General Arrangement and Outline Plan led to the decomposition of strategic planning into two stages: stowage location specification firstly by longitudinal position, and secondly by latitudinal position.

Longitudinal blocking of the cargo-space

Blocking cargo longitudinally by hatch means that a location of a container is specified only by hatch-lid (*i.e.* as being either above or below a particular hatch-lid). Typically this means specifying a number between 1 and, say, 12. This longitudinal specification is consistent with the degree of specification of container allocation provided in the General Arrangement. It is sufficient to ensure that crane usage is maximised - cargo can be spread across the ship into hatches such that all available cranes, separated by the necessary distances, can operate simultaneously. It is also sufficient to allow the bending and trim constraints to be determined to a reasonable degree of accuracy (taking into account the inexact statistical information used).

A physical data structure representing cargo space blocks was constructed (in LISP) which encoded necessary properties of the blocks.[10] These properties include restrictions on lengths of container and cargo type which can be accommodated, the number of external power supplies, and total TEU which can be stowed above and below deck. Also encoded were the semantic relationships between blocks (such as 'adjacent-to' and 'supported-by'), necessary for ensuring correct separation of hazardous cargo, and the minimising of costly hatch-lid movement.[10] The algorithm for performing search was constructed in the following way. A load-list is generated that, taking advantage of the generalised nature of the container descriptions in the statistical data used, is grouped into *classes* of containers[10] in a manner used by Shields.[9] Each class is made up of containers that share the same characteristics (destination, length) and is placed into ascending order of weight. Containers are then 'loaded' by specifying just a hatch number, sequentially, by using a search algorithm such as Branch-and-bound.[11] (A Simulated Annealing[2] approach can also be used here to load a number of containers at once into large block spaces at the start of the search, reducing the number as the spaces fill.)

This search is used to produce many different stowage configurations for a single port. In order to evaluate the effectiveness of the solutions, consideration must be given to stowage configurations at subsequent ports. A new pool of feasible solutions will be generated at each port-of-call for each of the arrival solutions. The number of solutions considered at each port is reduced with distance from the current port (and according to available time and processing resources) to avoid combinatorial increase of the state-space (indicated in Figure *4*). This is reasonable given the diminishing accuracy of statistical forecast data for distant ports.

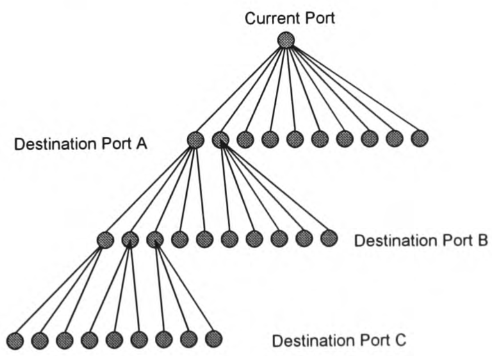


Figure 4 Branching factor reduced with distance

Branches are pruned from the state-space by taking advantage of problem constraints such as necessary separation of cargo types, bending moments and trim.

Each solution within the pool is then ranked according to the cost, in real terms, associated with crane utilisation, hatch-lid movement and the number of restows generated. The solution selected as best at the current port will be the one that is ranked highest when considering future ports, by simulating loading and unloading processes at each destination terminal.

Latitudinal blocking of the cargo-space

Given the containers allocated to each longitudinal block, the planning process can now distribute the containers within each longitudinal block between its corresponding latitudinal blocks. This degree of specification matches that indicated by Figure 3 and is consistent with that provided in the Outline Plan. At this stage, constraints relating to lateral distribution of weight and deck weight limits (*i.e.* torsion and heeling) can be determined to an acceptable level of tolerance using this model.

Each of the containers allocated to a given hatch will be allocated to a particular block within that hatch using a Branch and Bound search. Evaluation of each completed solution is based upon the requirements to maximise cargo-space usage, minimise hatch-lid movement, minimise over-stowage, and minimise heeling (on a local level). Constraints upon the type of container and cargo that can be placed in a given block, and on proximity of cargo-types, assist in reducing the state-space of the problem.

The most promising blocked outline-plan which satisfies intact stress and stability calculations, is selected for the tactical planning phase. (It is accepted by the authors that the use of ballast may be necessary in order to meet vessel stress and stability requirements, but it should be kept to a minimum - this should be reflected in the evaluation of a stowage pattern.)

3.3 The tactical planning document

A bay plan is a detailed view of just one of the stowage bays from the Outline Plan usually showing the above-deck and below-deck parts of the bay on separate sheets. A complete Bay Plan for a ship will be a large document composed of many sheets, each of which will be similar to the generic example shown in Figure 5.

The General Arrangement and Outline Plan are often used to indicate the broad allocation of groups of slots to containers of particular ports of discharge. The larger and more detailed Bay Plan is required for the planning and supervising of the actual stow for a loading operation and the detailed sequence for discharge. When planning is complete, each slot on the Bay Plan is labelled with information about the containers. This information includes the slot address, port of discharge and container identification code, type, dimensions, cargo content and weight. Non-containerised cargo and specials can also be indicated on the bay plan. In full, the bay plan contains the information required for the planners to make required intact stability calculations, and to ensure that maximum stack height and weight limits are not exceeded.

Voyage number:______ Date:______ Port:_____/______
Discharging/Loading

210814	210614	210414	210214	210014	210114	210314	210514	210714
210812	210612	210412	210212	210012	210112	210312	210512	210712
210810	210610	210410	210210	210010	210110	210310	210510	210710
210808	210608	210408	210208	210008	210108	210308	210508	210708
210804	210606	210406	210206	210006	210106	210306	210506	210706
	210604	210404	210204	210004	210104	210304	210504	
		210402	210202	210002	210102	210302		

Bay No. 21
Under deck
8' 6"

Figure 5 A Bay Plan

3.4 The tactical planning approach

The Tactical Planning phase uses the best solution found in the Strategic Planning phase, in which containers, described generally, are assigned only to blocks. The objective of this phase of planning is to refine the assignment so that specific containers are assigned to specific locations, as is the case in the Bay Plans. This will be achieved by obtaining a full stowage plan which is a starting point, and then altering that solution during an optimisation phase.

Obtaining an initial solution

A new loading algorithm is proposed here which is based upon three-dimensional *packing theory* (commonly applied to filling containers[3]) and using container-ship loading heuristics specific to this project. The heuristics were designed through an analysis of the stowage objectives outlined in Section 1 (the 'common-sense' derivation of which can be found in[10]). The principles underlying the loading heuristic are that: containers should not be stacked on top of others to be discharged earlier or, generally, above lighter ones; containers of the same length should be stacked together; over-width containers should be placed so as to minimise the number of slots which can no longer store containers due to the overlap of cargo (*i.e.* on the top positions of full stacks[10]).

For each block, the allocated containers are sorted according to size (largest to smallest standard containers, followed by out-of-gauge containers), destination (furthest first) and weight (heaviest first). The basic algorithm for loading is shown in Figure 6. Note that containers of standard and non-standard dimensions are dealt with separately.

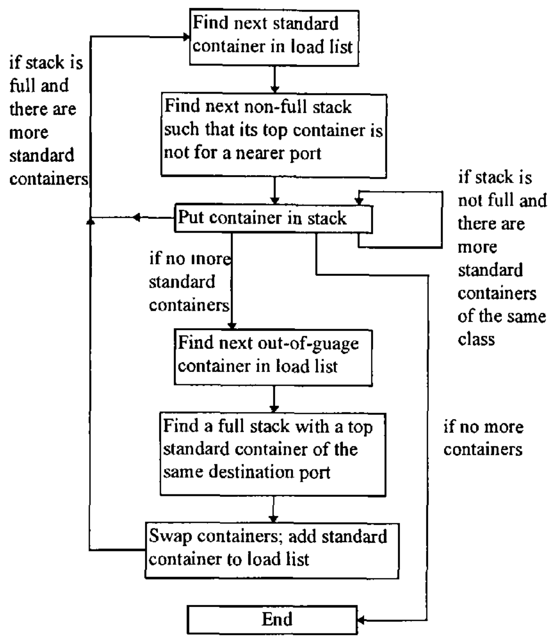


Figure 6 Algorithm for packing

The order in which the stacks of a block are filled depends on where the block is located (port, starboard or centre) so as to bias loading to the centre of the ship. This algorithm ensures that a stowage pattern can be found, for all blocks. Due to load-list order, this particular loading heuristic tends to minimise overstows, minimise void spaces, and produce stacks which are generally of containers of the same class. Also, heavier containers are generally stowed lower than lighter ones. The exact manner of loading can be varied, requiring changes to the above algorithm in order to encode different heuristics.[10] The heuristic used could, in itself, produce a good stowage solution. However, the solution is unlikely to be optimal, and may contain illegal relationships and poor location of containers with special requirements. The solution used at this stage, therefore, is used as a starting point for an optimisation process which rearranges the containers in the cargo space.

Optimisation

Human planners conceptually swap containers around until all constraints are satisfied and a near optimal solution is generated. A search methodology that models this approach would offer the most promise. Tabu search[4] was chosen.

Given a heuristically filled cargo-space, the task of the optimisation process is to rearrange the containers until such time as no further improvement is expected. The key to the process being the determination of a set (or *neighbourhood* [4]) of moves from the current state that are admissible. Due to the problem decomposition, the neighbourhood has been reduced to the set of permissible moves of containers within a single block of cargo-space. A move is simply the swapping of any two containers in that block. The state-space associated with a single block is small enough that a number of optimisation algorithms will, in theory, find an optimal solution. However, the Tabu search algorithm was selected for implementation and experimentation since it closely models the conceptual processes (where a stowage pattern is progressively altered by moving containers around) performed by the human planner.

The attractiveness of a valid stowage solution can be measured in a number of ways. A measure was chosen, to be incorporated into a single evaluation function, which values:

- a low number of overstows;
- a low number of stacks with mixed containers length;
- a low count of lighter containers stowed below heavier containers;
- a good distribution of weight across the width of the ship (as an indication of vessel stability).

4 Evaluation & conclusions

The combinatorial complexity of the deep-sea container-ship stowage problem is due to its large number of variables, such as vessel intact stability, hazardous cargo segregation and the need to distribute cargo intelligently to permit efficient manipulation at ports. There are a multitude of theoretically plausible solutions or stowage plans. An optimal solution can not be found for commercial sized ships in a reasonable processing time using commonly available computer software and hardware. However, the hybrid approach proposed in this paper, incorporating different AI techniques at each level of the decomposed model of the problem, appears to offer good sub-optimal results. For each sub-process, the state space becomes small enough to make the generation of these solutions feasible. Each level of the model allows checking of necessary constraints, to a satisfactory degree of accuracy, and closely models the processes used by human planners. Further, the approach allows more solutions to be considered than is possible for a human planner.

The longitudinal assignment is the largest of the sub-problems, but the abstraction used makes the application of general search theory plausible. It must be noted that its state space size will vary due to vessel capacity, specific cargo carried, and the number of future ports considered. Even so, it is expected that a solution, of acceptable quality, can always be generated in a viable length of time.

The other sub-processes have relatively small state-spaces. In latitudinal assignment, containers assigned to hatches are distributed between typically 16-20 blocks. In tactical planning, the quality of the initial solution is dependent on the heuristic employed, but will always be fast to determine as it only involves the placement of fewer than 100 TEUs. (A typical cargo-space block will hold approximately 12-60 TEUs.) Similarly, finding an optimum solution using Tabu search is also a relatively simple task. For below-deck blocks, which have restrictions on container lengths, experimentation on typical loads generated optimum solutions in very few iterations (as few as 15 iterations, and a recency list[4] of just one move). For on-deck blocks, this number increases due partly to variations in container lengths, but mostly due to the increased likelihood of hazardous cargo segregation requirements. However, in the worst cases, no more than 200 iterations, and recency lists of up to 7 moves, would be required.

The decomposition suggested, using generalised information at the start, and then the proposed form of optimisation using exact container details, lends itself well to the shipping operator. Most operators still accept containers for transport whilst the loading process is in progress and leaves the planning of bay-plans to the last moment, planning each bay in sequence as the precise container details become available.

5. Future Work

Former work in the area of full automation of the stowage planning problem failed to produce commercially viable approaches. This was due to attempts to reduce the problem in size and complexity until it fits the chosen hardware and software. The authors of this paper hope that the approach proposed here, in decomposing the problem without ignoring aspects of the full problem, will renew interest in the subject area. In particular, experimentation is required into four areas. Firstly, the implementation of Simulated Annealing for rapidly allocating containers to the longitudinal abstraction of cargo space in the strategic planning phase. Secondly, further experimentation is needed in the under-researched application of packing theory to container-ship cargo spaces. Thirdly, the experimentation with alternative neighbourhoods in Tabu search, such as increasing neighbourhood from blocks to hatches. Lastly, more experimentation with evaluation criteria is needed, with co-operation from Tonnage Centres, where the actual planning is performed.

References

[1] BOTTER, R. C., BRINATI, M. A. (1992). Stowage Container Planning: a Model for Getting an Optimal Solution, *IFIP Transactions B (Applications in Technology).*, Vol. B-5, pp271-29.

[2] DOWSLAND, K. A. (1995). *Modern Heuristic Techniques for Combinatorial Problems*, (Ed. Reeves, C.), Chapter 2: Simulated Annealing, McGraw-Hill, pp20-63.

[3] DOWSLAND, K. A., KNOTT, K., EGBELU, P. J. (1989). Packing problems, *European Journal of Operational Research*, Vol. 56, No. 1, pp2-14

[4] GLOVER, F., LAGUNA, M. (1995). *Modern Heuristic Techniques for Combinatorial Problems*, (Ed. Reeves, C.), Chapter 3: Tabu Search, McGraw-Hill, pp70-150.

[5] GOLDBERG, L. L. (1980). *Principles of Naval Architecture* (Ed. Comstock, J. P.), Chapter 2, pp63-42.

[6] ROACH, D. K., THOMAS, B. J. (1994). *Portworker Development Programme Unit C.2.2, Container-ship Stowage Plans* International Labour Organisation, ISBN 92-2-109271-2.

[7] ROSS, G (1996). Private communication, Maritime Computer and Technical Services Ltd, Cardiff.

[8] SATO, K., ITOH, H., AWASHIMA, Y. (1992). Expert System for Oil Tanker Loading/Unloading Operation Planning, *Computer Applications in the Automation of Shipyard Operation and Ship Design, VII* (Eds. C. Barauna Vieira et al.), Elsevier Science Publishers B.V. (North Holland).

[9] SHIELDS, J. J. (1984). Container-ship Stowage: A Computer-Aided Preplanning System, *Marine Technology*, Vol. 21, No. 4, pp370-383.

[10] WILSON, I. D. (1997). The Application of Artificial Intelligence Techniques to the Deep-Sea Container-ship cargo Stowage Problem. PhD Thesis (in preparation), University of Glamorgan.

[11] WINSTON, P. H. (1992). *Artificial Intelligence (3rd Edition)*, Addison-Wesley.